JOURNAL OF QUANTITATIVE LINGUISTICS, 2017
https://doi.org/10.1080/09296174.2017.1341724

Why One Cannot Estimate the Entropy of English by Sampling

Joachim von zur Gathen and Daniel Loebenberger

Bonn–Aachen International Center for Information Technology, Universität Bonn, Bonn, Germany

ABSTRACT

There have been attempts to approximate the entropy of English by frequency analysis of large corpora. Our original goal was to deduce more precise estimates by extensive calculations. This did not work well, thus confirming a widely held belief in linguistics. In order to put this belief on a firm basis, we used a simplified language model, closely related to others in the literature. This model exhibits an unexpected trichotomy: for very small n, say up to $n = 4$ in our case, n-gram counting is reasonably reliable; for medium n, up to 14, increasing statistical noise is added, and beyond that we see statistical noise only. The model is precise enough to yield explicit values for the thresholds given above dependent on the corpus size. Even though a mathematically rigorous proof for English itself is out of reach, our model gives a strong indication that frequency counting in (large) corpora is a dead end for approximating the entropy of English, and different linguistic tools and insights are required. As far as we know, this is the first rigorous quantifiable argument concerning the linguistic intuition that frequency counting of samples is insufficient for entropy determination.

1. Introduction

In the late 1940s, Claude Elwood Shannon defined the *entropy* of a probability distribution by an explicit formula involving the probabilities of the distribution and showed the importance of this concept in various areas. In his foundational paper on the subject, from 1948, he applied it to electronic communications and initiated the theory of error-correcting codes. The channel entropy determines the ultimate limits of their efficiency, namely their transmission rate. In 1949, he used it in cryptography. Here, the entropies of cleartexts and keys determine the ultimate limits of the ability to decipher encrypted messages without access to the secret key. In 1951, he studied the entropy of (printed) English and gave upper and lower bounds for it.

CONTACT Daniel daniel@bit.uni-bonn.de

Our interest comes from the second subject, namely cryptography. A particular case is the cryptanalysis of Vigenère ciphers from Kasiski (1863). Here, the frequencies of individual plaintext letters, letter-digrams, etc., survive with a flattened distribution; see von zur Gathen (2015, Section C.1). Many classical and modern encryption systems have been broken, but for short encrypted texts, the reliability of a decipherment depends critically on the entropy of the plaintext language, according to Shannon's *Unicity Bound*. Our original motivation was to find good bounds on the entropy of the distribution of letters and polygrams in English.

Since Shannon's 1951 work, linguists have tried to improve on his bounds; see for example Cover and King (1978) or Brown, Della Pietra, Mercer, Della Pietra, and Lai (1992). All of them have in common that a fairly large corpus is used to reflect on the true nature of English. We first follow this approach and describe in Sections 2 and 3 our calculations using the corpus of contemporary American English (COCA), see Davies (2008–2012), containing 450 million words of different categories of printed English, and determine some letter- and word-frequencies in it. Namely, for n between 1 and 30, we determine the frequency of letter-n-grams and the frequencies of word monograms. We can then apply Shannon's formula in two ways: either for a fixed n, we compute the entropy of, say, letter-n-grams, or we consider all letter-$(n-1)$-grams with the conditional frequency of the consecutive nth letter and compute the entropy of this distribution. The conditional entropy values calculated turn out to decrease with growing n.

However, although our corpus is presumably larger than those used earlier for this purpose, our numerical values for large polygram lengths were in conflict with known values and intuition. Indeed, for the analysis, ideally one would have a corpus of all texts in the language under discussion, printed English in our case. Then for each $n \geq 1$ and each letter-n-gram, one would determine its frequency. This yields a distribution over finite sequences of letters, to which we can apply Shannon's formula. However, no such corpus is available. This issue is well known in the linguistic community. During a fruitful discussion, Köhler (2016, private communication) expressed the following opinion on this:

> Zudem ist die Schützung der Entropie meines Wissens nur aufgrund eines unendlichen Strings möglich, wobei die Eigenschaften von Schützungen mittels abgebrochenem String fraglich sind. Au?erdem gibt es keine unendlich langen Texte. Andere Objekte wie Korpora sind künstlich zusammengestellt und entsprechen keiner linguistisch begründbaren Spracheinheit [Additionally, an estimate of the entropy is to our knowledge only possible when using an infinite string, whereas the properties of estimates employing truncated strings are questionable. Also, there are no infinitely long texts. Other objects, like corpora, are artificially assembled and do not correspond to linguistically justifiable language units].

Our experiments confirmed this intuition. However, we found that, for very small n, our frequency analyses lead to consistent results, whichever way a reasonably representative corpus was selected. It thus seems that we are indeed able to determine specific frequencies by analysing finite-sized corpora only, but we fail when considering larger n (and thus the entropy of English).

Specifically, we performed the computations on various derivatives of the corpus, say 26 letters plus space only, or only on extracts like texts labelled 'fictional' in the corpus. Our results are presented in Section 3. One finding is that the choice of derivative or sub-corpus does not influence the count substantially, thus showing a certain robustness of the sampling method. It would be interesting to see how other corpora fare in this respect.

One can argue that the frequencies obtained from a corpus are not representative, since a corpus is always a collection of (linguistic) objects whose statistical properties may have little in common with the set of all such objects. This is, of course, a valid point of view. The purpose of this work is, however, to show that sampling from a language cannot be used to estimate the entropy of the language reasonably. It turns out that, for the sampling method to provide reliable results, the required corpus size is completely out of reach.

Specifically, we noticed that from $n = 5$ on, there seems to be increasing statistical noise in our data on letter-polygrams, and for $n \geq 14$, the noise seems to dominate. This is, of course, a well-known behaviour and consistent with the above quotation from Köhler (2016, private communication). The main contribution of this work, starting in Section 5 and not tied to a particular language, is a precise analysis of this phenomenon. Namely, we describe a stochastic model for the entropy that explains this observation. Even though the language model we use is well known, we provide *explicit quantitative estimates* for the expected entropy in terms of the corpus and alphabet size, giving – at least for certain special cases – explicit bounds for those sizes that are necessary for good entropy approximations. As far as we know, such an explicit analysis has not been known before. Our results show that there is a trichotomy when analysing n-grams this way in any representative corpus: (1) reasonable approximations to the true value of the entropy for very small n; (2) the truth with some statistical noise for medium sized n; and (3) only statistical noise for large n. We conclude that the approach of bounding the entropy by analysing polygrams in a fixed (large) corpus is a dead end and cannot be carried much further than the current work. In order to get a better hold of a numerical value for the entropy of English, more linguistic insights are needed.

Our observations on the difficulty of approximating language entropy are consistent with results from the theory of computational complexity on this question, namely, that determining the entropy of a distribution is hard for a certain complexity class; see Section 5.

We concentrate our analysis on letter frequencies due to our cryptographic interest, but also do some computations with word frequencies. Here, the limitations discussed above show up even earlier, since our corpus has fewer words than letters.

The reader interested in only general conclusions may safely skip Sections 2 and 3, which deal with the English language, but mainly serve as a case study for our general findings in the later sections.

2. Description of the corpus

The corpus of contemporary American English (COCA), see Davies (2008–2012), consists of a large number of English texts from five different genres: academic texts; fictional texts; magazine texts; newspaper texts; and excerpts of spoken English. For the analysis, we considered only written English texts, i.e. we did not analyse the part of COCA that contains spoken English. One reason for this was that we did not succeed in removing artificially introduced tags (such as names) from the transcripts of spoken English, which might skew our statistical analyses. The resulting corpus consists of $2 \times 10^9 \approx 2^{31}$ characters and contains more than 340 million English words and roughly 65 million punctuation marks.

The characters in COCA are from the set of all 95 printable ASCII characters. These are classified as

- 26 lowercase Roman letters: `abcdefghijklmnopqrstuvwxyz`
- 26 uppercase Roman letters: `ABCDEFGHIJKLMNOPQRSTUVWXYZ`
- 10 Arabic numerals: `0123456789`
- 32 special symbols: `!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~`
- space: ␣

A first inspection shows that all but the special symbols `\^`{|~` occur in COCA. Special symbols are sometimes called *punctuation marks*.

Each of the above mentioned text genres are split into files containing corresponding texts from the years 1990 to 2012. Every file contains several articles which start with '`##`', followed by a seven digit identifier. Each article is split into paragraphs that are separated using a special HTML-type tag. For copyright reasons, the corpus is split into blocks of roughly 200 words to be compliant with the US Fair Use Law, 17 US Code ?107 through ?118, on copyrighted material.

For our analysis, we purged COCA of all article identifiers. We then replaced all sequences of Arabic numerals by the special symbol '`#`'. Furthermore, we substituted paragraph tags and block delimiters by one of the remaining unused symbols. These newly introduced special symbols are not directly used in our statistical analyses, but are used only to capture the properties of written English in a single block.

We chose to analyse COCA over the full alphabet of all printable ASCII characters first, distinguishing uppercase from lowercase Roman letters and keeping space and punctuation marks. A second analysis was done using the lowercase Latin alphabet with space only, that is, changing every Roman letter to its lowercase analogue while ignoring any punctuation but keeping space. Sequences of consecutive spaces were counted as a single space.

To distinguish the relevant cases, we used the following notions for the classification of certain types of ASCII characters.

- A *symbol* is any printable ASCII character.
- A *letter* is any lowercase Roman letter or space.
- A *string* is a sequence of symbols preceded and succeeded but not containing space.
- A *word* is a sequence of lowercase Roman letters preceded and succeeded by space.

A single occurrence of one of the above defined notions is also called a *monogram*. For a fixed $n \geq 1$, we call the *polygram* containing n consecutive monograms an *n-gram*.

The purged COCA thus resembles excerpts of written English containing a large number of string-monograms, separated by space, each containing an arbitrary number of symbols (excluding space). All punctuation marks and contractions such as `n't`, `'re`, `'s` or the Saxon genitive `'s` are monograms.

By ignoring all punctuation marks, replacing each uppercase Roman letter by its lowercase analogue and substituting any sequence of consecutive spaces by a single space, we obtain the corresponding corpus for word-monograms, where the words are also separated by space. Both corpora can then be used for the statistical analysis.

3. Elementary statistical analyses

After purging, we counted the occurrences of symbol-, letter-, string- and word-monograms. For any of these choices M of the set of monograms, we then computed the frequency distribution of n-grams. Such a frequency distribution simply counts how often a certain n-gram, say $g \in M^n$, occurs in the corpus. Dividing this count by the number of n-grams considered gives the *probability* $p_n(g)$ that the n-gram g occurs.

Definition 3.1.

(1) *The* Shannon entropy *of the distribution* p_n *is*

$$H(p_n) = -\sum_{g \in M^n} p_n(g) \log_2 p_n(g).$$

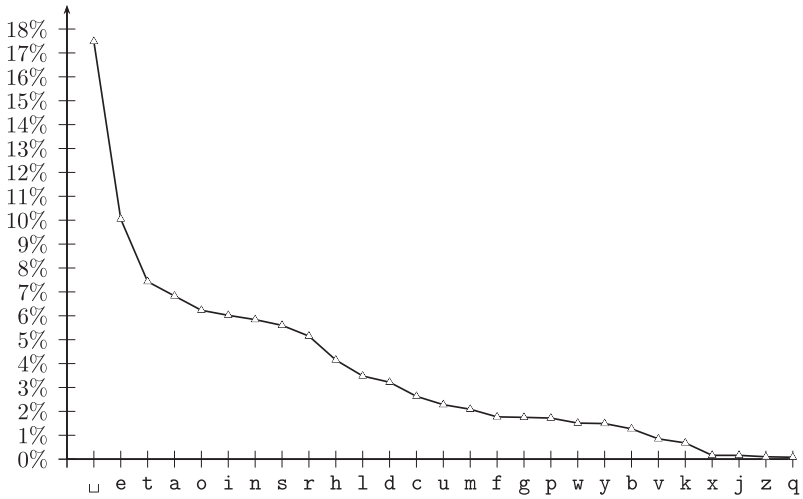

Figure 1. Frequencies of letter-monograms over the alphabet containing lowercase Latin letters and space only.

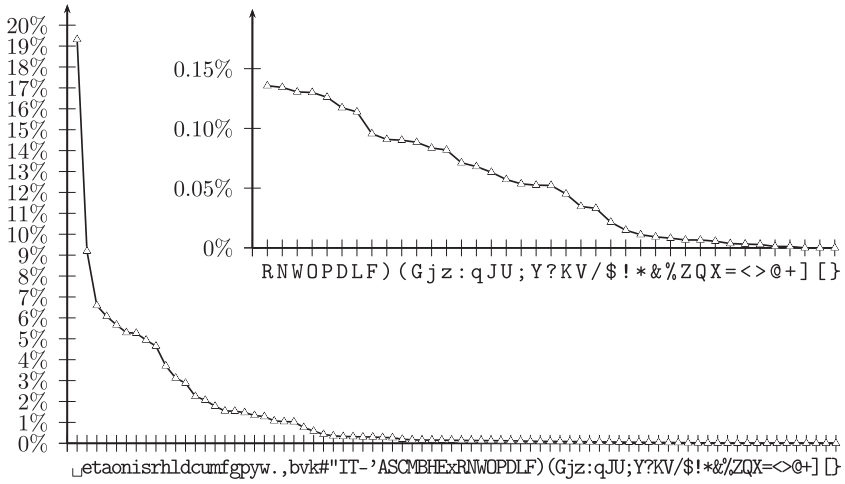

Figure 2. Frequencies of symbol-monograms over the alphabet of all printable ASCII characters. Characters that do not occur in COCA are not plotted.

(2) *The* conditional entropy *of the distribution* p_n *given* p_{n-1} *is*

$$H(p_n : p_{n-1}) = H(p_n) - H(p_{n-1}).$$

The latter definition corresponds to the chain rule for the entropy and can be found in any textbook on information theory such as that by Cover and Thomas (2006).

Most of the following computational experiments were carried out on a 2.2 GHz Intel Core i7 with 8 GB RAM. For larger computations, we employed a small cluster with 8 dual-core 3.00 GHz Intel Xeon CPUs and 64 GB RAM. For all of the following numerical results, we provide only graphs in the text; see Appendix 1 for listings of the underlying numerical data.

Table 1. The most frequent word-monograms.

Monogram	the	of	and	to	a	in	that	s	for	i
Frequency (%)	5.72	2.74	2.73	2.52	2.34	1.89	1.10	0.97	0.89	0.85

Table 2. The most frequent nouns in COCA.

Monogram	time	people	years	way	year	world	day	life
Frequency (‰)	1.55	1.30	1.12	0.96	0.82	0.74	0.74	0.69

Table 3. The most frequent letter-digrams.

Digram	th	he	in	er	an	re	on	at	en	nd
Frequency (%)	10.19	9.32	7.78	6.47	6.20	5.57	4.96	4.50	4.30	4.06

3.1. Monogram frequencies

The letter-monogram and the symbol-monogram frequencies of the purged COCA can be found in Figures 1 and 2, respectively. In both statistics, space '␣' is by far the most frequent character, followed by a number of lowercase Latin letters. From the figures, we directly observe that, for both alphabets considered, the ten most frequent letters constitute more that 70% of all characters. From Definition 3.1(1) we directly obtain a letter-entropy[1] of 4.12 *bits* and a symbol-entropy of 4.46 *bits*. In COCA, we have the most frequent word-monograms given in Table 1. In this table, the contraction `'s` and the Saxon genitive `'s` are counted as the single word-monogram `s`. For the frequency distribution, we obtain a word-entropy of 11.05 *bits*. Concerning string-monograms, we observe that the punctuation symbol '`,`' occurs most often, followed by '`.`' and some of the above most frequent word-monograms from Table 1. As a side remark, we also list the most frequent nouns from COCA in Table 2.

3.2. Polygram frequencies

Concerning polygram frequencies, we first analysed the most frequent letter-digrams in COCA (see Table 3).

For the letter-digram entropy, Definition 3.1(1) yields 7.55 *bits*, and for the symbol-entropy, 8.01 *bits* per digram. This gives by Definition 3.1(2) a conditional letter-digram entropy of 3.43 *bits* and a conditional symbol-entropy of 3.56 *bits*.

For larger n, we obtain for the conditional letter-n-gram entropy the values given in Figure 3.

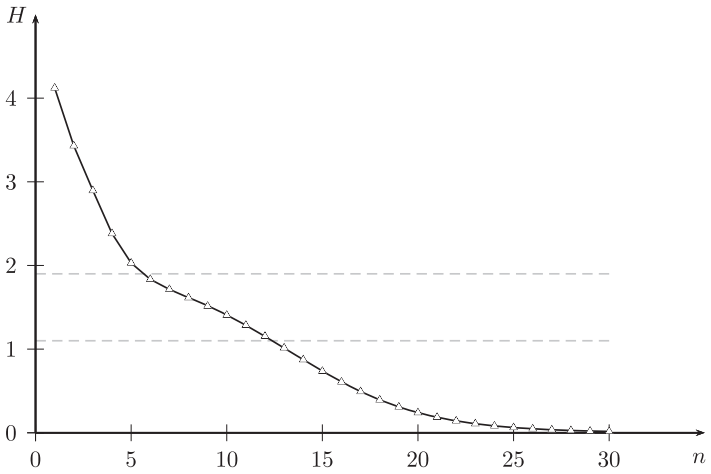

Figure 3. Conditional letter-n-gram entropy of COCA for $n = 1, \ldots, 30$. The dashed grey lines are Shannon's 1951 bounds for the entropy of English.

The figure indicates that the sampling errors introduced by considering the frequency distribution of successive n-grams grow as n gets larger. An intuitive explanation of this behaviour might be the following: a fixed corpus of length l is for growing n a decreasingly representative sample. Eventually, we arrive at a value of n for which each of the $l - n + 1$ occurring n-grams appears exactly once in the corpus and we get absolute entropy $\log_2(l - n + 1)$. But then, each of the $l - n$ occurring $(n + 1)$-grams also appears only once in the corpus and we get relative entropy $\log_2(1 + 1/(l - n)) \approx 1/(l - n)$, which is close to zero for the relevant choices for l and n.[2]

The left-hand part of Figure 3 seems to indicate an English entropy around 1.5 bits, but this is pure guesswork. The only conclusion we can definitely draw is that the entropy is below three – presumably a rather poor estimate. For a more detailed analysis of what happens, see Section 5.

One challenge in the statistical analysis of COCA was actually to fit the frequency distributions of polygrams into computer memory: the corpus consists of roughly $2^{31} \approx 10^{21.49}$ characters. If we only *stored* all n-grams of the corpus in memory (actually to start the analysis), we would need approximately $n \times 2^{31}$ bytes, which starts getting impractical already for $n = 4$. Thus, one either needs to refrain from the idea of storing the data in memory (but use slow hard-drives instead) or develop an approach that uses a certain amount of memory that does not grow so fast with n. This can be achieved by considering the number of different n-grams in the corpus. Consider a given set of monograms M with $k = \#M$ letters. Then there are at most k^n different n-grams in the corpus. For our alphabets, we have $k = 27$ letters and $k = 95$ (more precisely, $k = 89$ as explained above) different symbols.

(1) For small n, the number of different n-grams is comparatively small. Also, one expects for these choices of n that almost each n-gram occurs.

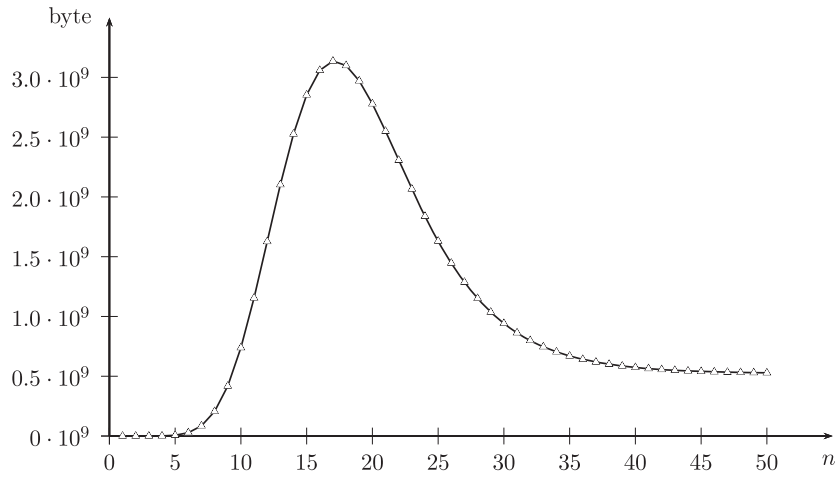

Figure 4. Storage requirements for the repeated *n*-grams over the alphabet of lowercase Latin letters and space in COCA for successive *n*.

(2) For very large *n*, we have an enormous amount of different *n*-grams. However, most of them either do not occur at all or just once.

Trading between these two extremes, the idea is to track *repeated* *n*-grams only, that is, *n*-grams that occur at least twice in the corpus. Then one expects for both small *n* and large *n* that the number of stored repeated *n*-grams is not too large. This in turn leads to the following algorithm: we read the corpus monogram by monogram and store for each monogram the position at which the monogram occurs. We then recursively use the frequency distribution of repeated $(n-1)$-grams to compute the frequency distribution of repeated *n*-grams. In this algorithm, each position is stored using a 4 byte unsigned integer. We thus store for the full COCA with approximately 2×2^{30} positions roughly $8 \times 2^{30} = 8\,GB$ in memory. Additionally, we have to store each occurring repeated *n*-gram. In Figure 4, the storage requirements for the occurring repeated *n*-grams are plotted.

From the figure, we see that the storage requirements grow monotonically for *n* from 1 to 17 and decrease afterwards monotonically. At maximum, we have to store roughly $3.13 \times 10^9 \approx 2.92 \times 2^{30}$ bytes. Thus, the whole algorithm requires at most 10.92 GB of memory, which fits into decent hardware such as our small Intel Xeon cluster.

The occurrences of repeated *n*-grams deviate from what one would initially suspect. For growing *n*, the *number* of repeated *n*-grams tends to zero. However, one would expect this to happen much earlier than here. Indeed, there are whole parts of sentences that occur several times in the corpus and are thus counted repeatedly. For example, the 16-gram `indeed␣there␣are`, which is quite frequent in academic texts and incidentally also the beginning of the previous sentence, occurs 135 times in COCA and is also counted 135 times.

Further problems in the statistical analysis of COCA are inherent problems with the precision of the floating point arithmetic for computing the entropy. Since, by Definition 3.1(1), the entropy is computed as a sum of k^n summands of the form $p \log_2 p$, tiny errors in the evaluation of the log-function may accumulate and lead to large errors in the evaluation of the entropy value. To circumvent this, we decided to use `mpfr`, a C library for multiple-precision floating-point computations with correct rounding; see Fousse, Hanrot, Lefévre, Pélissier, and Zimmermann (2007). Specifically, the use of arbitrary-precision arithmetic enabled us then to evaluate the entropy correctly.

4. Reproducibility of the results

We explain now how readers of the current work who are so inclined can reproduce our results for written English, but also generalize the given methodologies for other written languages. The restriction to written languages is based on the definition of language as 'a set (finite or infinite) of sentences, each finite in length and constructed out of a finite set of elements', see Chomsky (1957).

This is the basis of all our analyses: we start from a finite set of elements (as described in Section 2, these are Roman letters with some special punctuation marks in the case of written English) and perform statistical computations as described in Section 3.

To reproduce our results for written English, one needs a source of samples from the language. As explained in the introduction, we decided to use in our case Davies' COCA corpus, but any other source will do as well. From the selected source, all 1-grams, 2-grams, etc. are extracted and the corresponding probabilities (and possibly the positions) of the occurrences are stored. Then, one can use this as a basis for computing several relevant values, such as the (conditional) entropy as given in Definition 3.1. Of course, other metrics, such as for example the repetition rate, can be computed as well. The results of the computation will exhibit the following behaviour: when considering n-grams for growing n, the quality of the statistical results gets worse and worse. This is intrinsic to the sampling methodology as we will show in the subsequent sections.

The techniques can be used for other (written) languages as well. For languages using a 'small' alphabet, say with up to 100 characters and for which large corpora are available, the methods used for written English can be used as such and a comparison with the results for English should be easily possible. It will be interesting to compare with languages having a large alphabet (such as Chinese), or a small corpus (such as Rongorongo or ancient Egyptian).

Similarly, one can extend our results to a phonetic representation of a language. There, the set of basic elements would be the phonetic characters and the computations would then be performed over concatenations thereof.

Both generalizations are beyond the scope of this treatise.

We will now show how we can model formally any kind of language using basic elements in the sense of Chomsky. Although our motivation comes from experiments with the English language, our arguments apply to a large class of languages.

5. A stochastic model for natural language entropy

It is well known that when we consider the language under consideration as a stationary random process

$$X = (\dots, X_{-2}, X_{-1}, X_0, X_1, X_2, \dots),$$

over a finite set M of $k \in \mathbb{N}_{\geq 1}$ monograms, the entropy of the process X is defined as

$$H(X) = \lim_{n\to\infty} H(X_n : X_0, \dots, X_{n-1}).$$

If the language is an ergodic process, then for any n-gram $(x_0, \dots, x_{n-1}) \in M^n$, we have by the Shannon–McMillan–Breiman theorem (see for example Algoet & Cover, 1988) almost surely (over the choice of $(x_0, \dots, x_{n-1})$)

$$H(X) = \lim_{n\to\infty} -\frac{1}{n} \log_2 \left(prob((X_0, \dots, X_{n-1}) = (x_0, \dots, x_{n-1}))\right), \quad (1)$$

Thus, one can compute the entropy of such a process by looking at sufficiently long samples and computing the relative entropy of the distribution of successive n-grams. We have the following well-known result.

Fact 5.1. *For a stationary ergodic stochastic process X, $H(X_n : X_0, \dots, X_{n-1})$ is non-increasing in n and has a limit $H'(X)$.*

***Proof*:** By assumption

$$H(X_n : X_0, \dots, X_{n-1}) \leq H(X_n : X_1, \dots, X_{n-1}) = H(X_{n-1} : X_0, \dots, X_{n-2}),$$

since X is stationary. Thus $H(X_{n-1} : X_0, \dots, X_{n-2})$ is non-negative and non-increasing and has a limit $H'(X)$. □

This is consistent with our observations depicted in Figure 3. By the chain rule we have $H(X_{n-1} : X_0, \dots, X_{n-2}) = H(X_0, \dots, X_{n-1}) - H(X_0, \dots, X_{n-2})$ since X is stationary. Since X is ergodic, one might want to approximate it as in Equation (1) by looking at sufficiently many examples.

We will show in the following that the amount l of text one needs for precise computations of the entropy of the language is too large to be feasible. Thus, any such approach can in principle only illuminate a limited part of the linguistic truth. To complete the picture, we also show how large a corpus needs to be *at most* for expectedly precise entropy computations.

Our observations are consistent with results from the theory of computational complexity on this question. Namely, Goldreich, Sahai, and Vadhan (1999) show that determining the entropy of a distribution is hard for the complexity class NISZK of non-interactive statistical zero-knowledge. Here, a program tries to approximate the entropy of a distribution on about 2^n elements by just asking for samples according to the distribution; together these samples make up a corpus. Their result says that (under usual complexity-theoretic assumptions) it is infeasible to obtain good approximations to the entropy.

5.1. Description of the model

As above, we consider the language as a strongly stationary ergodic stochastic process $X = (\dots, X_{-2}, X_{-1}, X_0, X_1, X_2, \dots)$ over the set of $k \geq 1$ monograms $M = \{m_1, \dots, m_k\}$. To simplify our analysis, we additionally assume in our model that for some $n \in \mathbb{N}_{\geq 1}$ the probability for the occurrence of a specific monogram only depends on the previous n letters. In other words, we model X as a homogeneous nth-order Markov process. This is a frequently used stochastic model for English, a nice survey of other possible models can be found in Rosenfeld (2000).

For our analysis we are interested in the n-grams that come from X. Thus, we define a second process $X^{\bar{n}} = (\dots, X^{\bar{n}}_{-2}, X^{\bar{n}}_{-1}, X^{\bar{n}}_0, X^{\bar{n}}_1, X^{\bar{n}}_2, \dots)$ of n-grams, where for each $i \in \mathbb{Z}$ we define $X^{\bar{n}}_i = (X_i, \dots, X_{i+n-1})$. The process $X^{\bar{n}}$ is now by construction a first-order homogeneous Markov process. Thus, there are for any $x, y \in M^n$ (unknown) transition probabilities $T_n : M^n \times M^n \to \mathbb{R}$ for the process $X^{\bar{n}}_i$ induced by the language considered that specify the probability $T(x, y)$ of occurrence of a certain n-gram x given that the previous n-gram was y. Thus $T(x, y) = 0$ unless x and y overlap in all but one letter.

The stationary distribution $S_n(x)$ of the process $X^{\bar{n}}$ is the probability that a certain n-gram is observed, and defined as

$$\begin{aligned} S_n(x) &= prob(X^{\bar{n}}_i = x) \\ &= \sum_{y \in M^n} prob(X^{\bar{n}}_{i-1} = y) prob(X^{\bar{n}}_i = x : X^{\bar{n}}_{i-1} = y) \\ &= \sum_{y \in M^n} prob(X^{\bar{n}}_{i-1} = y) \cdot T_n(x, y) \end{aligned}$$

for $x \in M^n$. This distribution is well-defined if the underlying Markov process is *irreducible* and *recurrent*. This assumption seems to hold for English, and we will take it for granted in the following.

We define the *observed distribution* (in information theory also called the *type*) of the n-grams induced by the process X over a set of values $dom(X) = M$ when observing $l \in \mathbb{N}_{\geq 0}$ consecutive outcomes by

$$p_n^l(X): \begin{array}{rl} dom(X)^n & \longrightarrow \frac{1}{l}\mathbb{Z}, \\ x & \longmapsto \frac{1}{l}\#\{0 \leq i < l; X_i^{\bar{n}} = x\}. \end{array} \tag{2}$$

Thus, for $x \in dom(M^n)$, we have $p_n^l(X) = \frac{1}{l}\sum_{0\leq i<l} \mathbb{1}_{X_i^{\bar{n}}=x} =$, where $\mathbb{1}_{X_i^{\bar{n}}=x}$ is the indicator function of the predicate $X_i^{\bar{n}} = x$, that is, $\mathbb{1}_{X_i^{\bar{n}}=x} = 1$ if the ith n-gram in the process $X^{\bar{n}}$ is $x \in M^n$, and $\mathbb{1}_{X_i^{\bar{n}}=x} = 0$ otherwise. The observed distribution $p_n^l(X)$ is a random variable with values in the finite set

$$P_n^l(X) = \{p: dom(X)^n \longrightarrow \frac{1}{l}\mathbb{Z} : p_n^l(X) = p\} \tag{3}$$

of all possible observable distributions induced by corpora of length l.

The problem is now to estimate how far the conditional entropy $H(p_n^l(X) : p_{n-1}^l(X)) = H(p_n^l(X)) - H(p_{n-1}^l(X))$ of the observed distribution differs from the conditional entropy $H(S_n : S_{n-1})$ of the stationary distribution. Suppose we have $|H(p_{n-1}^l(X)) - H(S_{n-1})| \leq \varepsilon_{n-1}$ and $|H(p_n^l(X)) - H(S_n)| \leq \varepsilon_n$ for some $\varepsilon_{n-1}, \varepsilon_n > 0$. Then

$$|H(p_n^l(X) : p_{n-1}^l(X)) - H(S_n : S_{n-1})| \leq \varepsilon_{n-1} + \varepsilon_n$$

by the triangle inequality. In other words, it is sufficient to estimate when the observed entropies $H(p_{n-1}^l(X))$ and $H(p_n^l(X))$ differ only slightly from the true entropies $H(S_{n-1})$ and $H(S_n)$, respectively, in order to be able to deduce corresponding results for the conditional entropy. We will thus restrict our attention to the entropy only.

It is easy to establish an upper bound on the observed entropy $H(p_n^l(X))$. Because log is a concave function, the entropy in Definition 3.1(1) attains its maximum if $p_n^l(X)$ is a uniform distribution. Since there are in total $\#M^n = k^n$ possible n-grams and we consider exactly l consecutive n-grams, we obtain the upper bound

$$H(p_n^l(X)) \leq \min(n\log_2(k), \log_2(l)). \tag{4}$$

We now analyse the behaviour of the expectation $E\left(H(p_n^l(X))\right)$. Our primary goal is to establish a lower bound on l for which the difference $|E\left(H(p_n^l(X))\right) - H(S_n)|$ is bounded from below. This in turn leads to the conclusion on how large we have to select the corpus size l at least to be able to approximate the correct value of the entropy with small error. The second

goal is to provide an upper bound on l giving an appropriate upper bound on $|E\left(H(p_n^l(X))\right) - H(S_n)|$. Having this allows us to conclude how large a corpus has to be *at most* for a useful entropy approximation.

By the definition of the expectation of a random variable, we have

$$E\left(H(p_n^l(X))\right) = \sum_{p \in P_n^l(X)} H(p) prob(p_n^l(X) = p). \tag{5}$$

Before we consider this expression in full generality, we first discuss a special case which is easy to analyse. Afterwards we will argue that a similar reasoning also holds for arbitrary distributions.

5.2. Randomspeak

We now restrict ourselves to the special case that S_n is the uniform distribution U_{M^n} on M^n, i.e. for $x \in M^n$, we have $S_n(x) = U_{M^n}(x) = 1/k^n$. We know that $H(S_n) = \log k^n$, but now ask how this value can be approximated by observations on corpora of some length l. Indeed, in this case the desired bounds on l can be derived. Consider the *necessary* size of l first. By (4) we have $H(p_n^l(X)) \leq \min(n \log_2(k), \log_2(l))$ and by (5) also

$$E\left(H(p_n^l(X))\right) \leq \min(n \log_2(k), \log_2(l)). \tag{6}$$

Thus, $\log_2 k^n - E\left(H(p_n^l(X))\right) \geq \log_2(k^n/l)$. In order to approximate the true value $\log_2 k^n$ with relative error at most $\alpha > 0$, we consider the inequality $\log_2(k^n/l) \leq \alpha \log_2 k^n$ and solve for l, giving

$$l \geq k^{(1-\alpha)n}. \tag{7}$$

This says, for example, that when we want to approximate the n-gram entropy with relative error at most $\alpha = 0.05$ over an alphabet with $k = 27$ letters, COCA does not provide sufficient data about the n-gram entropy for $n > 6$. If we wanted to say something about 10-grams, we would already need a corpus with at least 36 TB of text. A corpus of the storage size used by all of humankind[3] would let us look until $n = 15$, but even this does not provide enough data for $n > 15$.

We will now analyse which corpus size l is *sufficient* for good entropy approximations by sampling only. Consider a distribution $p\colon M^n \to \mathbb{R}$ which is close to uniform. More specifically, assume that the statistical distance is bounded by $\delta \in \mathbb{R}_{>0}$ so that $\|p - U_{M^n}\|_\infty < \delta$. Then by definition of the max-norm, we have for all $g \in M^n$ that $|p(x) - k^{-n}| < \delta$, that is, $p(x) \in [k^{-n} - \delta, k^{-n} + \delta]$. Consequently, $-\log_2 p(x) \in [-\log(k^{-n} + \delta), -\log(k^{-n} - \delta)]$ and $-p(x)\log_2 p(x) \in [-(k^{-n} - \delta)\log_2(k^{-n} + \delta), -(k^{-n} + \delta)\log_2(k^{-n} - \delta)]$.

We thus obtain the lower bound

$$\begin{aligned} H(p) &= -\sum_{x\in M^n} p(x)\log_2 p(x) \\ &\geq -(1-\delta k^n)\log_2(k^{-n}+\delta). \end{aligned}$$

Thus, we have for the expected value

$$\begin{aligned} E\left(H(p_n^l(X))\right) &= \sum_{p\in P_n^l(X)} H(p)prob(p_n^l(X)=p) \\ &\geq \sum_{\substack{p\in P_n^l(X)\\ \|p-U_{M^n}\|_\infty<\delta}} H(p)prob(p_n^l(X)=p) \\ &\geq -(1-\delta k^n)\log_2(k^{-n}+\delta) \sum_{\substack{p\in P_n^l(X)\\ \|p-U_{M^n}\|_\infty<\delta}} prob(p_n^l(X)=p) \\ &= -(1-\delta k^n)\log_2(k^{-n}+\delta)prob(\|p_n^l(X)-U_{M^n}\|_\infty<\delta). \end{aligned} \tag{8}$$

Without loss of generality, assume that n divides l. Otherwise, pad the corpus accordingly. Since $p_n^l(X)(x) = \frac{1}{l}\sum_{0\leq i<l}\mathbb{1}_{X_i^{\bar{n}}=x}$, the idea is to split for $x\in M^n$ the relative counts $p_n^l(X)(x)$ into n independent parts, that is, consider for $0\leq j<n$ the relative counts

$$p_{n,j}^l(X)(x) = \frac{1}{l}\sum_{\substack{0\leq i<l\\ i=j \text{ in } \mathbb{Z}_n}} \mathbb{1}_{X_i^{\bar{n}}=x} = \frac{1}{l}\sum_{0\leq i<\frac{l}{n}} \mathbb{1}_{X_{in+j}^{\bar{n}}=x},$$

which are the occurrences of x at positions that fall into residue class j modulo n and then apply Hoeffding's inequality to get a bound for $prob(\|p_n^l(X)-U_{M^n}\|_\infty<\delta)$.

Recall that Hoeffding's inequality states that when we have l independent random variables $X_0,\ldots,X_{l-1}$ such that almost surely $a_i\leq X_i-E(X_i)\leq b_i$, then for all positive real constants $\varepsilon\in\mathbb{R}_{\geq 0}$ we have

$$prob\left(\left|\sum_{0\leq i<l}(X_i-E(X_i))\right|\geq\varepsilon\right)\leq 2\exp\left(\frac{-2\varepsilon^2}{\sum_i(b_i-a_i)^2}\right).$$

Let $0\leq j<n$ and set $Y_i=\mathbb{1}_{X_{in+j}^{\bar{n}}=x}$ for $0\leq i<l/n$, then the random variables $Y_0,\ldots,Y_{l/(n-1)}$ are independent and $E(Y_i)=k^{-n}$ for all i. Furthermore,

we have $-k^{-n} \le Y_i - E(Y_i) \le 1 - k^{-n}$ and Hoeffding's inequality gives for any $x \in M^n$ and $\varepsilon > 0$

$$\begin{aligned} prob\left(\left|p^l_{n,j}(X)(x) - \frac{1}{n}k^{-n}\right| \ge \frac{\varepsilon}{l}\right) &= prob\left(\left|\frac{1}{l}\sum_{0\le i<l/n} \mathbb{1}_{X^{\bar{n}}_{in+j}=x} - \frac{1}{n}k^{-n}\right| \ge \frac{\varepsilon}{l}\right) \\ &= prob\left(\left|\sum_{0\le i<l/n} Y_i - \sum_{0\le i<l/n} E(Y_i)\right| \ge \varepsilon\right) \\ &= prob\left(\left|\sum_{0\le i<l/n} (Y_i - E(Y_i))\right| \ge \varepsilon\right) \\ &\le 2\exp\left(\frac{-2n\varepsilon^2}{l}\right). \end{aligned}$$

Setting $\delta = \varepsilon/l$, we get

$$prob\left(\left|p^l_{n,j}(X)(x) - \frac{1}{n}k^{-n}\right| \ge \delta\right) \le 2\exp(-2n\delta^2 l).$$

Note that $p^l_n(X)(x) = \sum_{0\le j<n} p^l_{n,j}(X)(x)$. By the triangle inequality we have

$$\begin{aligned} &prob\left(\left|p^l_n(X)(x) - k^{-n}\right| \ge n\delta\right) \\ &\le prob\left(\sum_{0\le j<n}\left|p^l_{n,j}(X)(x) - \frac{1}{n}k^{-n}\right| \ge n\delta\right) \\ &\le prob\left(\exists\, 0 \le j < n : \left|p^l_{n,j}(X)(x) - \frac{1}{n}k^{-n}\right| \ge \delta\right) \\ &\le n \cdot prob\left(\left|p^l_{n,0}(X)(x) - \frac{1}{n}k^{-n}\right| \ge \delta\right) \\ &\le 2n\exp(-2n\delta^2 l). \end{aligned}$$

We obtain

$$\begin{aligned} prob\left(\|p^l_n(X) - U_{M^n}\|_\infty \ge \delta\right) &= prob\left(max_{x\in M^n}|p^l_n(X)(x) - k^{-n}| \ge \delta\right) \\ &\le \sum_{x\in M^n} prob\left(|p^l_n(X)(x) - k^{-n}| \ge \delta\right) \\ &\le 2k^n n\exp(-2n\delta^2 l). \end{aligned} \tag{9}$$

Plugging this into (8) gives

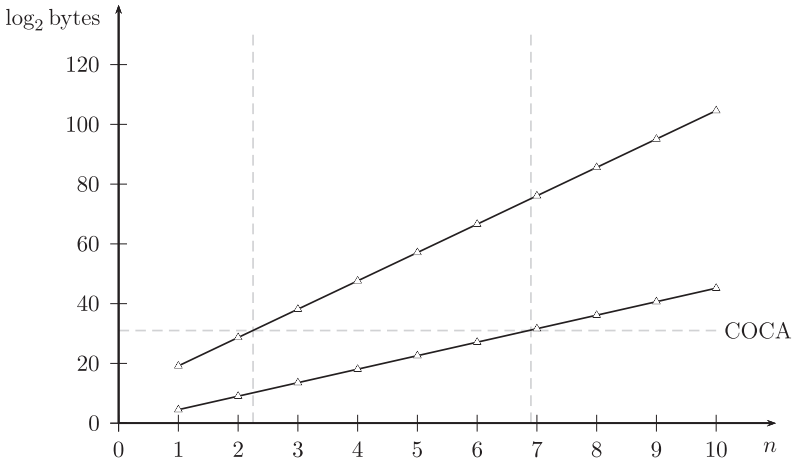

Figure 5. Logarithmic scale lower and upper bounds on the corpus size *l* for approximating the entropy of randomspeak with relative error $\alpha = 0.05$ over an alphabet with 27 letters, instantiating (7) and (10). The horizontal grey dashes show the size of COCA and the vertical dashes pass through its intersections with the lower and upper bounds, respectively.

$$E\left(H(p_n^l(X))\right) \geq -(1-\delta k^n)\log_2\left(k^{-n}+\delta\right)(1-2k^n n\exp\left(-2n\delta^2 l\right)). \tag{10}$$

This says, for example, that when we want to approximate the n-gram entropy with relative error $\alpha = 0.05$ and $\varepsilon = 0.05$ over an alphabet with $k = 27$ letters, COCA tells us only for sure a good approximation on the entropy for $n \leq 2$. A corpus of the storage size used by all of humankind, i.e. 295 exabyte, definitely provides sufficient data for $n \leq 6$. We summarize our results in Figure 5.

Thus, we have satisfactory results in the case of randomspeak: first, it is infeasible to approximate the entropy by looking at increasingly long n-grams. Second, the amount of text we need to look at *at most* is enormous and a corpus of size of COCA can serve as a basis for estimating the n-gram entropy for $n = 1$ and $n = 2$, since for these values the corpus size sufficient for good approximations lower than COCA's size. For $n = 3, \ldots, 6$ we do not know whether COCA is sufficiently large, but we know it is larger than what is necessary for a good entropy approximation by sampling. For $n > 6$, sampling cannot be used to estimate the entropy, since the necessary size of a corpus exceeds that of COCA.

We are able to obtain the same result numerically, using the data given in Table A5. When we interpolate the data linearly by the equation

$$y = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1) + y_1,$$

for points (x_1, y_1) and (x_2, y_2) on the line, we obtain that the lower bound is approximately given by the line equation

$$y = 4.517143127 \cdot x$$

and the upper bound is approximately given by

$$y = 9.49987258 \cdot x + 9.66950108,$$

where we use for both extrapolations the corresponding points at $x_1 = 1$ and $x_2 = 2$, respectively. COCA's size is roughly 2^{31} bytes, which tells us that the upper bound is larger than the horizontal line at $y = 31$ for $x > 2.24$ and the lower bound for $x > 6.86$. This is consistent with our deduction based on (7) and (10).

5.3. Markov sampling

We will now argue that for non-uniform stationary distributions we also have the stated trichotomy. In fact, it seems that randomspeak is the worst case that can happen when sampling.

For the lower bound this is easy to see. In fact, we can proceed similarly as in the beginning of Section 5.2, but this time we do not assume anything about the Markov transition probabilities T_n (and thus the stationary distribution S_n). From (6), we know that we have for the expected entropy $E\left(H(p_n^l(X))\right) \leq \min\left(n \log_2(k), \log_2(l)\right)$. Thus, $H(S_n) - E\left(H(p_n^l(X))\right) \geq H(S_n) - \log_2 l$. In order to approximate the true value $H(S_n)$ with relative error at most $\alpha > 0$, we consider the inequality $H(S_n) - \log_2 l \leq \alpha H(S_n)$ and solve for l, giving $l \geq 2^{(1-\alpha)H(S_n)}$. Since $H(S_n) \leq n \log_2 k$, this bound is indeed weaker than the corresponding bound in (7) and thus says that. For non-uniform S_n, a smaller corpus is necessary for a good approximation of the entropy of English than for the case of randomspeak.

It remains to argue that this is also true for the *necessary* corpus size. Thus, one has to study the difference between randomspeak and a Markov process with unknown (possibly non-uniform) transition probabilities T_n. In order to do so, we used successive approximations to English. Specifically, we computed for every $m \geq 1$ from COCA the frequency of letter m given the previous $m - 1$ letters and generated equally long texts randomly corresponding to the respective distributions. This well-known procedure gives for $m = 1$ exactly randomspeak, while for growing m the resulting random language from the $(m + 1)$th-order Markov process approaches English better and better.

Afterwards, we computed for all of the generated texts the n-gram entropy values for successive n and plotted the result, see Figure 6.

The figure shows that in the case $m = 1$, i.e. randomspeak, the conditional entropy value is maximal for very small n and decreases rapidly. When m

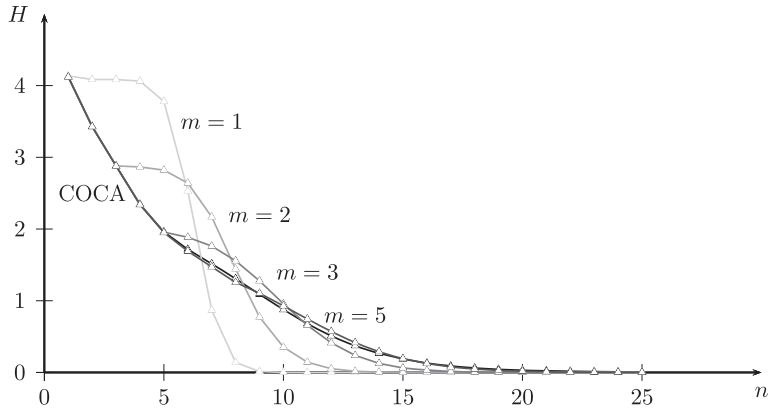

Figure 6. Letter entropy of a (1/128)-fraction of COCA (black) versus letter entropy of zeroth-order (light grey), first-order, second-order and fourth-order approximations (dark grey).

grows, the behaviour gets more and more similar to that of COCA, where we have a much slower decrease in the entropy values for growing m, thus giving conditional entropy zero much later. This leads to the conclusion that statistical noise in the case of English occurs somewhat later than in the case of uniform distributions.

6. The central conjecture

We have proven mathematically that there is a natural trichotomy in the case of randomspeak when analysing n-grams by sampling: reasonable approximations to the true value of the entropy for very small $n \leq 2$, the truth with some statistical noise for medium sized $2 < n < 7$, and statistical noise only for large $n \geq 7$. We also argued that in the stochastic model a similar trichotomy holds in general and saw that the case of randomspeak is the worst case possible. The result is difficult to quantify, since the entropy of English and thus the specific bounds for the necessary and sufficient corpus size, respectively, are unknown. That this is also true for English (regardless of the model) leads to the following central conjecture of this article.

Conjecture 6.1. *The approximation of the n-gram entropy of English by sampling corpora leads to a natural trichotomy: (1) the linguistic truth for very small n; (2) the truth with some statistical noise for medium size n; and (3) only statistical noise for large n.*

If true, we further conjecture that it holds for other languages with 'small' alphabets and 'large' corpora, if they follow an irreducible recurrent Markov process.

We performed further experiments to underpin this conjecture. The idea was to analyse how the entropy of a representative fraction of COCA differs from the entropy value of the full corpus. Our trichotomy conjecture implies

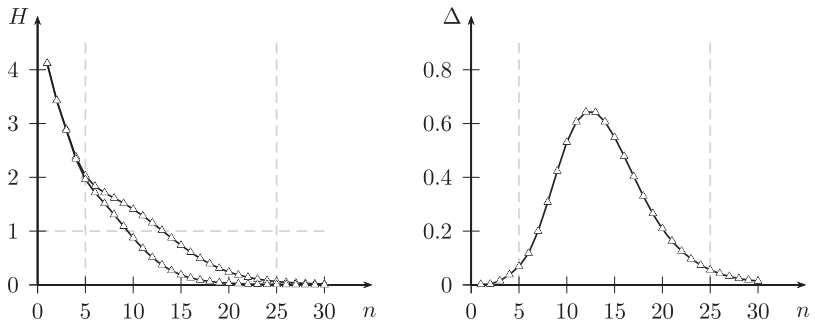

Figure 7. Illustration of the trichotomy. Left: letter-entropy of the full COCA (the upper) and a (1/128)-fraction thereof (the lower). Right: absolute distance Δ between the two entropy values.

that we expect almost no difference for very small and very large n, since in the former case we computed in both cases a good approximation to the true value and in the latter case we anyway have in both cases only statistical noise. The results of such an analysis are depicted in Figure 7.

The figure indicates that, in the case of English, we have a good approximation to the true value of the n-gram entropy for $n \leq 5$. For $n = 14$ the measured n-gram entropy drops below one for the first time, which means that the statistical noise seems to dominate from this point in time on, resulting in only noise beyond $n \geq 25$. This observation is also consistent with our observation above, which led us to the conclusion that randomspeak is in fact the worst case that can happen.

7. Bounding the expected entropy

We finish by giving bounds on the expectation (5) of the entropy in our Markov model described in Section 5.1. Recall that it is defined as

$$E\left(H(p_n^l(X))\right) = \sum_{p \in P_n^l(X)} H(p) prob(p_n^l(X) = p).$$

First, let us compute the probability that an observed sequence $(X_0^{\bar{n}}, \ldots, X_l^{\bar{n}})$ of n-grams is equal to a fixed given one. To do so, we use the distribution $p_2^l(X^{\bar{n}})$ of consecutively occurring n-grams, i.e. the bigram distribution of the process $X^{\bar{n}}$. We have for $(x_0^{\bar{n}}, \ldots, x_{l-1}^{\bar{n}}) \in (M^n)^l$ with $prob(X_0^{\bar{n}} = x_0^{\bar{n}}) \neq 0$:

$$\begin{aligned} &\frac{1}{prob(X_0^{\bar{n}} = x_0^{\bar{n}})} prob((X_0^{\bar{n}}, \ldots, X_{l-1}^{\bar{n}}) = (x_0^{\bar{n}}, \ldots, x_{l-1}^{\bar{n}})) \\ &\qquad = \prod_{1 \leq i < l} prob(X_i^{\bar{n}} = x_i^{\bar{n}} : X_{i-1}^{\bar{n}} = x_{i-1}^{\bar{n}}) \qquad (11) \\ &\qquad = \prod_{x,y \in M^n} T_n(x,y)^{l \cdot p_2^l(x^{\bar{n}})(x,y)} \end{aligned}$$

$$= \prod_{x,y \in M^n} 2^{l \cdot p_2^l(x^{\bar{n}})(x,y) \log_2 T_n(x,y)}$$
$$= 2^{-l \cdot \left(H\left(p_2^l(x^{\bar{n}}) \parallel T_n\right) + H(p_2^l(x^{\bar{n}}))\right)}, \tag{12}$$

where we write

$$H\left(p_2^l(x^{\bar{n}}) \parallel T_n\right) = \sum_{x,y \in M^n} p_2^l(x^{\bar{n}})(x,y) \log_2 \frac{p_2^l(x^{\bar{n}})(x,y)}{T_n(x,y)}$$

for the *conditional entropy* (also called the *Kullback–Leibler divergence* or *information gain*) of $p_2^l(x^{\bar{n}})$ given T_n; see Kullback and Leibler (1951). The result of (12) is the Markov analogue of a well-known result for independent draws; see for example Cover and Thomas (2006, Section 11.1).

Using (12), we can compute the probability that an observed distribution of consecutive n-grams $p_2^l(X^{\bar{n}})$ equals a given distribution $q \in P_2^l(X^{\bar{n}})$. Assuming that the first n-gram of the corpus was drawn uniformly at random, i.e. $prob(X_0^{\bar{n}} = x_0) = 1/k^n$, we have

$$\begin{aligned} prob(p_2^l(X^{\bar{n}}) = q) &= \sum_{\substack{x \\ p_2^l(x^{\bar{n}}) = q}} prob((X_0^{\bar{n}}, \ldots, X_{l-1}^{\bar{n}}) = (x_0^{\bar{n}}, \ldots, x_{l-1}^{\bar{n}})) \\ &= \sum_{\substack{x \\ p_2^l(x^{\bar{n}}) = q}} prob(X_0^{\bar{n}} = x_0) \cdot 2^{-l \cdot \left(H\left(p_2^l(x^{\bar{n}}) \parallel T_n\right) + H(p_2^l(x^{\bar{n}}))\right)} \\ &= \frac{1}{k^n} c_q \cdot 2^{-l \cdot (H(q \parallel T_n) + H(q))}, \end{aligned} \tag{13}$$

writing $x = (\ldots, x_{-2}, x_{-1}, x_0, x_1, x_2, \ldots)$ for a specific outcome of the process X and c_q for the number of such sequences with $p_2^l(x^{\bar{n}}) = q$. We have the following result in our context.

Lemma 7.1. *Let* $c_q = \#\{x = (\ldots, x_{-2}, x_{-1}, x_0, x_1, x_2, \ldots);\ p_2^l(x^{\bar{n}}) = q\}$. *Then*

$$\frac{1}{(l+1)^{k^{n+1}}} k^n 2^{lH(q)} \leq c_q \leq k^n 2^{lH(q)}.$$

***Proof*:** This can be proved as in Cover and Thomas (2006, Theorem 11.1.4) by using (13) for the result of the probability of a certain distribution in the context of Markov processes, noting that we have at most k^{n+1} non-zero values in the distribution q. □

The lower bound in the lemma can be substantially improved. In fact, one can replace the constant $(l+1)^{k^{n+1}}$ by the much smaller $\#P_2^l(Y^{\bar{n}})$, where $Y^{\bar{n}}$

is the Markov process with transition probabilities given by $T_n = q$. Jacquet, Knessl, and Szpankowski (2012) gave asymptotic estimates for this count when $n = 2$. They proved that, up to a constant, we have asymptotically $\#P_2^l(Y^{\bar{2}}) \approx l^{k^2-k}/(k^2-k)!$. The constant they give is dependent on the alphabet size k and expressed as a certain multi-integral. We are not aware of a similar result for arbitrary n. This might also be due to the fact that the behaviour of overlapping strings is quite subtle. Guibas and Odlyzko (1981) analysed this issue and gave fundamental results on the number of strings without a specified pattern. This should give better bounds on $\#P_2^l(Y^{\bar{2}})$, but since we do not need this for the following, we stick to the lemma as stated above.

It remains to express the expected entropy (5) of $p_n^l(X)$ in terms of the entropy of the bigram distributions of its n-grams. By the chain rule and noting that the entropy of p differs from the entropy of the marginal distributions of q by at most a factor of 2 we have

$$\begin{aligned} E\left(H(p_n^l(X))\right) &= \sum_{p \in P_n^l(X)} H(p) prob(p_n^l(X) = p) \\ &\in \left[\frac{1}{4} \dots 1\right] \sum_{q \in P_2^l(X^{\bar{n}})} H(q) prob(p_2^l(X^{\bar{n}}) = q). \end{aligned} \tag{14}$$

Combining (13) and Lemma 7.1, we thus get the following bounds on the expected entropy:

$$E\left(H(p_n^l(X))\right) \in \left[\frac{1}{4 \cdot (l+1)^{k^{n+1}}} \dots 1\right] k^n \sum_{q \in P_2^l(X^{\bar{n}})} H(q) \cdot 2^{-lH(q \,\|\, T_n)}. \tag{15}$$

This seems to be difficult to handle in its full generality for the following reasons.

- Both the transition probabilities T_n of the Markov process and the corresponding stationary distribution S_n from which the samples are taken are unknown.
- Computing the exact number of sequences with a given bigram distribution of n-grams is out of reach at the moment.
- The conditional entropy is not a metric. Specifically, it does not satisfy the triangle inequality.

8. Conclusion

We performed a thorough analysis of Davies' corpus of contemporary American English and computed the entropy values for various alphabets and n-gram lengths. After observing that this gives results incompatible with known ones, we studied why sampling cannot be used for estimating the entropy

of English in a satisfactory manner, since the size of the corpus necessary is beyond practical limits. To show this, we set up a simplified Markov model for a natural language like English and argued that sampling procedures for n-grams can only give reasonable approximations of the entropy for very small n and no result at all for large ones, leading to a natural trichotomy. Although our mathematical analysis applies to the artificial language randomspeak, we conjecture that, regardless of the model, this trichotomy also applies to English and other languages with similar properties (sizes of alphabets and corpora, Markovian generation, as stated above), and give experimental results to validate this hypothesis.

The fundamental conclusion is that linguistic methods different from our style of computational analysis of orthographic representations are needed to understand the entropy of English.

Notes

1. The entropy of the distribution of lowercase Roman letters *without space* is 4.19.
2. In the purged COCA, we have $l \approx 2^{31}$. Even if only for $n > 2^{30}$ each letter n-gram occurs only once, we still get relative letter entropy $\log_2 (1 + 2^{-30}) \approx 2^{-30} \approx 0$.
3. Following Hilbert and López (2011), the storage used nowadays is estimated as 295 exabyte.

Acknowledgements

The authors are grateful to Graeme Hirst for advice on the subject and to Reinhard Köhler for enlightening discussions.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

The authors' work was funded by the Bonn–Aachen International Center for Information Technology foundation (the B-IT Foundation) and the state of Nordrhein-Westfalen.

References

Algoet, P. H., & Cover, T. M. (1988). A sandwich proof of the Shannon-McMillan-Breiman theorem. *The Annals of Probability, 16*, 899–909.

Brown, P. F., Della Pietra, V. J., Mercer, R. L., Della Pietra, S. A., & Lai, J. C. (1992). An estimate of an upper bound for the entropy of English. *Computational Linguistics, 18*, 31–40. Retrieved from http://www.aclweb.org/anthology/J92-1002

Chomsky, N. (1957). *Syntactic structures*. The Hague/Paris: Mouton Publishers.
Cover, T. M., & King, R. C. (1978). A convergent gambling estimate of the entropy of English. *IEEE Transactions on Information Theory, 24*, 413–421. Retrieved from https://protect-us.mimecast.com/s/LL62BWsGEwgoc6?domain=www-isl.stanford.edu
Cover, T. M., & Thomas, J. A. (2006). *Elements of information theory* (2nd ed.). Hoboken, NJ: Wiley.
Davies, M. (2008–2012). The corpus of contemporary American English: 450 million words, 1990-present. Retrieved from http://corpus.byu.edu/coca/
Fousse, L., Hanrot, G., Lefévre, V., Pélissier, P., & Zimmermann, P. (2007). MPFR: A multiple-precision binary floating-point library with correct rounding. *ACM Transactions on Mathematical Software, 33*(2), 13:1–13:15. doi:10.1145/1236463.1236468
Goldreich, O., Sahai, A., & Vadhan, S. (1999). Can statistical zero knowledge be made noninteractive?, or On the relationship of SZK and NISZK. In M. Wiener (Ed.), *Advances in cryptology: Proceedings of CRYPTO 1999 Santa Barbara, CA*, Vol. 1666 (pp. 467–484). Berlin: Springer-Verlag. doi:10.1007/3-540-48405-130
Guibas, L. J., & Odlyzko, A. M. (1981). String overlaps, pattern matching, and nontransitive games. *Journal of Combinatorial Theory, Series A, 30*, 183–208. doi:10.1016/0097-3165(81)90005-4
Hilbert, M., & López, P. (2011). The world's technological capacity to store, communicate, and compute information. *Science, 332*, doi:10.1126/science.1200970
Jacquet, P., Knessl, C., & Szpankowski, W. (2012). Counting Markov types, balanced matrices, and Eulerian graphs. *IEEE Transactions on Information Theory, 58*, 4261–4272. doi:10.1109/TIT.2012.2191476
Kasiski, F. W. (1863). *Die Geheimschriften und die Dechiffrir-Kunst*. Berlin: E. S. Mittler und Sohn.
Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics, 22*, 79–86.
Rosenfeld, R. (2000). Two decades of statistical language modeling: Where do we go from here? *Proceedings of the IEEE, 88*, 1270–1278. doi:10.1109/5.880083
Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal, 27*, 379–423 and 623–656. ; Reprinted in Claude E. Shannon and Warren Weaver, The Mathematical Theory of Communication, University of Illinois Press, Urbana, IL, 1949.
Shannon, C. E. (1949). Communication theory of secrecy systems. *Bell System Technical Journal, 28*, 656–715.
Shannon, C. E. (1951). Prediction and entropy of printed English. *Bell System Technical Journal, 30*, 50–64. Retrieved from https://protect-us.mimecast.com/s/kJYMBRfW0J7bcl?domain=princeton.edu
von zur Gathen, J. (2015). *Cryptoschool* (p. 32). Heidelberg: Springer-Verlag.

Appendix 1. Numerical results

For better reproducibility of our results, we list here the numerical results of our findings. Specifically, we give the numerical data underlying each statistical plot in the current work.

Table A1. Numerical data for Figure 1.

Letter	Percentage	Letter	Percentage	Letter	Percentage
␣	17.48	e	10.05	t	7.43
a	6.83	o	6.23	i	6.02
n	5.84	s	5.6	r	5.15
h	4.15	l	3.48	d	3.22
c	2.63	u	2.28	m	2.09
f	1.77	g	1.75	p	1.72
v	0.85	k	0.68	x	0.16
j	0.16	z	0.1	q	8.0e−2

Table A2. Numerical data for Figure 2.

Symbol	Percentage	Symbol	Percentage	Symbol	Percentage
␣	19.31404379287664	e	9.189638118956339	t	6.5890425217861575
a	6.071616515232053	o	5.663258901190674	n	5.295936607767364
i	5.267291123777249	s	4.939274100379097	r	4.648566422939864
h	3.7036267300235646	l	3.1249643771273696	d	2.8781581619544947
c	2.2443613703617467	u	2.061698754700811	m	1.774177196473276
f	1.5459613552911255	g	1.5361174456573872	p	1.4682669809940119
y	1.3316968772841433	w	1.2773286682033456	.	1.073121149502952
,	1.0361571023906162	b	1.0283952281531639	v	0.7568996712682905
k	0.5871472114118468	#	0.4274953355446141	"	0.35747406873944276
I	0.3252573828772203	I	0.3200413882158496	-	0.3048304097480939
’	0.3033881872242249	A	0.27768224815272513	S	0.26768550987540857
C	0.1967761014338056	M	0.16410211509418077	B	0.15352246570438938
H	0.15218280860281624	E	0.15044607205311294	x	0.14771990873428092
R	0.13568836352143035	N	0.1343510773265215	W	0.13050546671709276
O	0.13010212807536894	P	0.12610017988049896	D	0.1172865240286478
L	0.11384676522200697	F	9.554108970352677e−2	)	9.076323859371124e−2
(	9.013343094741738e−2	G	8.835036688149446e−2	j	8.357597729540875e−2
z	8.196309690984636e−2	:	7.11950552765129e−2	q	6.807498952448084e−2
J	6.331710144877836e−2	U	5.722557837442999e−2	;	5.3581684503996446e−2
Y	5.247262178458918e−2	?	5.227066795492757e−2	K	4.5178622268261734e−2
V	3.461775045925126e−2	/	3.311355243517853e−2	$	2.1521478481615078e−2
!	1.4812666148160127e−2	*	1.1032208053865239e−2	&	9.217136747974818e−3
%	8.103379633372326e−3	Z	6.618623043946523e−3	Q	6.6059624023593775e−3
X	5.58054527006719e−3	=	3.6104640864674897e−3	<	3.123716948295036e−3
>	2.8255043080644897e−3	@	1.1965491753184307e−3	+	1.1435357023055906e−3
]	1.801889064837139e−6	[	6.164357327074423e−7	}	9.483626657037574e−8

Table A3. Numerical data for Figure 3.

n	Entropy	n	Entropy	n	Entropy
1	4.122315434233340347702778672 1	2	3.430760333026745101392407377 72	3	2.895406035830700552935468294 89
4	2.381383692515017358459772367 5	5	2.029537310205823175124351109 84	6	1.834326001346735068864290951 75
7	1.714458630715927967003153753 47	8	1.614492914175357896056084427 98	9	1.514326050290417668975351261 91
10	1.406173268165776590876703266 99	11	1.285839340218153381556476233 53	12	1.154379364532729823622503317 89
13	1.015337744974388556329358834 77	14	0.873801964361653915602801134 81	15	0.736241419930706797458697110 415
16	0.608101337413167186696227872 744	17	0.492909765357492091197855188 511	18	0.393127959487284073247792548 5
19	0.309379594892263298788748215 884	20	0.240705964776896763623881270 178	21	0.185644837339676627152584842 406
22	0.142317516604322236162261106 074	23	0.108512498820687142142560333 014	24	0.082443522791944445771150640 212
25	0.062540181374576064854409196 4141	26	0.047455883861147185598383657 6343	27	0.036048338269623059204604942 3516
28	0.027470935851965805341023951 7689	29	0.021028972765805065137101337 3137	30	0.016174311609354674601490842 1785

Table A4. Numerical data for Figure 4.

n	Byte	*n*	Byte	*n*	Byte
1	27	2	1456	3	53787
4	861416	5	6442885	6	27780660
7	85127511	8	205537080	9	418009311
10	738908360	11	1155084634	12	1626795432
13	2102850516	14	2526503210	15	2853373740
16	3058867728	17	3136688632	18	3099008322
19	2970835060	20	2778861960	21	2549987811
22	2307096836	23	2066215902	24	1838391504
25	1630647350	26	1446641768	27	1287056007
28	1151009720	29	1036822268	30	941963970
31	863460794	32	799113120	33	746578932
34	703882144	35	669317215	36	641373948
37	618802763	38	600526426	39	585757302
40	573754120	41	564009202	42	556096002
43	549656057	44	544449532	45	540244035
46	536847646	47	534095357	48	531887664
49	530129775	50	528773400		

Table A5. Numerical data for Figure 5. Lower and upper bounds are given in $\log_2$ bytes.

n	Lower bound	Upper bound
1	4.517143127	19.16937366
2	9.034286254	28.66924624
3	13.55142938	38.14079409
4	18.06857251	47.62002396
5	22.58571564	57.10622396
6	27.10285876	66.59742953
7	31.62000189	76.09221661
8	36.13714502	85.58962923
9	40.65428814	95.08901879
10	45.17143127	104.5899337

Table A6. Numerical data for Figure 6.

n	COCA	$m = 1$	$m = 2$
1	4.120449529593618542833155515 84	4.135120418173469758471583190 84	4.119322048265736668781755724 92
2	3.427809916539334977869657450 36	4.085342961459538457802409539	3.426100669445911073296429094 63
3	2.880129216636876599011429789 24	4.084369347646147119235138234 22	3.420909085927889670131207822 12
4	2.343392547017254301522370951 72	4.061909473711509122040297370 4	3.410694050513631836452077550 35
5	1.960920706897573140281565429 18	3.778574303233060760476291761 73	3.325090331434834567403413529 97
6	1.716923144179723692559491610 15	2.531408500260269534010149072 86	2.913730004253508809597406070 68
7	1.514409070238652788020772277 38	0.869003396347569179170022835 024	1.941715131088724177743642940 19
8	1.307099133048346573104936396 7	0.139856753325481975025468273 088	0.853372380810309749676889623 515
9	1.091100346532002163257857318 97	0.013244996333725111981038935 4825	0.244935380779132572115486254 916
10	0.875671274150846556949545629 323	0.001033420839242182864836649 96922	0.050252937896225091662927297 8753
11	0.679524529219101225407939637 08	7.665439828485887119313701987 27e−05	0.008224923570278264151056646 3694
12	0.511316545567048308384983101 86	6.792279375389398410334251821 04e−06	0.001184519294454844384745229 03562
13	0.373625521296688134498253930 36	0	0.000162295223226749385503353 55103
14	0.267256742742095099174548522 569	0	2.210681510916856495896354317 67e−05
15	0.187627243286527800592011772 096	0	0
16	0.129999598483248490765618043 952	0	0
17	0.089963122124601113682729192 0781	0	0
18	0.062421302908710174506268231 1982	0	0
19	0.043648220321770736518374178 5586	0	0
20	0.031000441297006631202748394 5712	0	0
21	0.022318023669502196071334765 2927	0	0
22	0.016192488171999031010273029 0964	0	0
23	0.011887685365646660784477717 2431	0	0
24	0.008894153616616051749588223 17421	0	0
25	0.006724529221639130582843790 76213	0	0

(*Continued*)

Table A6. (*Continued*).

n	$m = 3$	$m = 5$	$m = 8$
1	4.120273370014228220270524616 35	4.120250479639285146049587638 12	4.120584835399499823438418388 83
2	3.427181918514148328824830969 1	3.427975527310123560198462655 53	3.427854063032757636619862751 1
3	2.880416999488988771815911604 79	2.880079164822949344682001537 88	2.879118948010523482139433326 67
4	2.864907874255484543368766026 11	2.341313050355605440699946484 53	2.339484085194579066069309192 24
5	2.821364529924926145554309186 99	1.956816032733039278923570236 66	1.950130474194805785259632102 68
6	2.640578588761187006639374885 71	1.884660873851080609142627508 85	1.690900958681229582225569174 62
7	2.169168509664110899848310509 7	1.761944982148058613802277250 22	1.472614836320417452952824532 99
8	1.443337319966325082987168570 98	1.557289887276059658915983163 75	1.255427796172206456049025291 58
9	0.773337444331957613030681386 59	1.276967655968196169169459608 38	1.104725865884017110829518060 2
10	0.353603474783280802284934907 22	0.957322469970261380467491107 993	0.929110114182229551715863635 764
11	0.143457724337178404994119773 619	0.657164387070061195572634604 05	0.746178808152073713699792278 931
12	0.052149183147680133743051555 939	0.413876476202432286299881525 338	0.572127930826376029926905175 671
13	0.017129184399927055437728995 4573	0.238180961674775204528486938 216	0.416873584505289329626975813 881
14	0.005187797505154634336577146 31408	0.126534325249661350198948639 445	0.290519185026049342468468239 531
15	0.001445309803823846550585585 64633	0.062798795627710291000767028 8906	0.193808797932589982337958645 076
16	0.000380733052438131380768027 156591	0.029220836930942795106602716 2597	0.124231302805920762466485030 018
17	0.000102197427725769784956355 79197	0.012829218743778625366758205 9279	0.077704814899213658918597502 6339
18	2.612150247927047530538402497 77e−05	0.005451869533182218674483010 54537	0.047331987667309505241064471 1927
19	4.510447254801874805707484483 72e−06	0.002220754521911061371433962 430805	0.028433099849308973716688342 3924
20	1.258086214761533483397215604 78e−06	0.000899196016820980048578348 942101	0.017079620607422185685209115 0358
21	0	0.000349213750073573692134232 260287	0.010392216328877879050196497 7011
22	0	0.000132356099665997817282914 184034	0.006342003831019127346735331 22987
23	0	5.130589646995531438733451068 4e−05	0.003966581116127798622983391 3967
24	0	2.310300430607981070352252572 77e−05	0.002560472709600247753769508 56298
25	0	8.432829812221598331234417855 74e−06	0.001682356077179036901725339 70326

Table A7. Numerical data for Figure 7.

n	COCA	(1/128)-COCA	Distance
1	4.122315434233340347702778672 1	4.120449529593618542833155515 84	0.0018659
2	3.430760333026745101392407377 72	3.427809916539334977869657450 36	0.00295042
3	2.895406035830700552935468294 89	2.880129216636876599011429789 24	0.0152768
4	2.381383692515017358459772367 5	2.343392547017254301522370951 72	0.0379911
5	2.029537310205823175124351109 84	1.960920706897573140281565429 18	0.0686166
6	1.834326001346735068864290951 75	1.716923144179723692559491610 15	0.117403
7	1.714458630715927967003153753 47	1.514409070238652788020772277 38	0.20005
8	1.614492914175357896056084427 98	1.307099133048346573104936396 7	0.307394
9	1.514326050290417668975351261 91	1.091100346532002162632578573 1897	0.423226
10	1.406173268165776590876703266 99	0.875671274150846556949545629 323	0.530502
11	1.285839340218153381556476233 53	0.679524529219101225407939637 08	0.606315
12	1.154379364532729823622503317 89	0.511316545567048308384983101 86	0.643063
13	1.015337744974388556329358834 77	0.373625521296688134498253930 36	0.641712
14	0.873801964361653915602801134 81	0.267256742742095099174548522 569	0.606545
15	0.736241419930706797458697110 415	0.187627243286527800592011772 096	0.548614
16	0.608101337413167186696227872 744	0.129999598483248490765618043 952	0.478102
17	0.492909765357492091197855188 511	0.089963122124601113682729192 0781	0.402947
18	0.393127959487284073247792548 5	0.062421302908710174506268231 1982	0.330707
19	0.309379594892263298788748215 884	0.043648220321770736518374178 5586	0.265731
20	0.240705964776896763623881270 178	0.031000441297006631202748394 5712	0.209706
21	0.185644837339676627152584842 406	0.022318023669502196071334765 2927	0.163327
22	0.142317516604322236162261106 074	0.016192488171999031010273029 0964	0.126125
23	0.108512498820687142142560333 014	0.011887685365646660784477717 2431	0.0966248
24	0.082443522791944445771150640 212	0.008894153616616051749588223 17421	0.0735494
25	0.062540181374576064854409196 4141	0.006724529221639130582843790 76213	0.0558157
26	0.047455883861147185598383657 6343	0.005129632443097165150902583 26381	0.0423263
27	0.036048338269623059204604942 3516	0.004025454632319735992496134 71329	0.0320229
28	0.027470935851965805341023951 7689	0.003210710935594107695578713 90134	0.0242602
29	0.021028972765805065137101337 3137	0.002532070250722995297110173 85125	0.0184969
30	0.016174311609354674601490842 1785	0.002006817559692564145734650 08289	0.0141675