# Willy vs. Jimmy: Which Typing Monkey gets all the Bananas?

Neil M Hennessy

January 2003

**Abstract**

The typing monkeys scenario postulates that, given enough time, the works of William Shakespeare could be produced as the result of a random process acting on a typewriter. The original scenario relies on calculations from classical probability. The newly postulated scenario stages a literary competition amongst typing monkeys by applying methods from information theory and algorithmic complexity theory in order to judge a representative work of William Shakespeare against a representative work of James Joyce.

The classic typing monkeys scenario is concerned with the amount of time an industrious monkey would take to produce part or all of the works of William Shakespeare. W. J. ReMine argues that The Bard's works could not be produced by a whole typing pool of monkeys:

> The monkeys could not randomly type merely the first 100 characters of *Hamlet*. If we count only lowercase letters and spaces (27 characters in all), then the probability of typing the 100 characters is one chance in $27^{100}$ (one chance in $1.4 \cdot 10^{143}$). If each proton in the observable universe were a typing monkey (roughly $10^{80}$ in all), and they typed 500 characters per minute (faster than the fastest secretary), around the clock for 20 billion years, then all the monkeys together could make $5 \cdot 10^{96}$ attempts at the 100 characters. It would require an additional $3 \cdot 10^{46}$ such universes to have an even chance at success. We scientifically conclude that the monkey scenario cannot succeed. For the scientist it would be perverse to insist otherwise [1].

The likelihood that a monkey would produce *Hamlet* has been likened to the likelihood of the phenomena to which we refer as life being produced as the result of random processes. In 1602, Willy the Typing Monkey entered *Hamlet* into the Stationer's Register of England, much to the chagrin of the calculators of probability. Turning our attention from the production of a literary work to its reception, we are interested in how to measure Willy's achievement.

If we consider the first 100 characters of Hamlet, and 50 repetitions of the characters ‘ab’, then according to the probabilistic calculations above, they are equally likely to emerge from the typing monkey pool:

> ‘the tragedie of hamlet actus primus scoena prima enter barnardo and francisco two centinels barnardo’
>
> ‘ababababababababababababababababababababababababab’

Continuing with the assumption that the monkey types at random, the chance of typing the first ‘a’ is 1/27, the chance that the next character is ‘b’ is 1/27, etc. up to the last character. The probability of the entire sequence being produced as the result of random typing is 1 chance in $27^{100}$, as before. The probability calculations do not distinguish between one of the greatest works of literature, and a sequence of repeated ‘ab’s.

When Willy turns in *Hamlet*, and another monkey from the pool submits pages and pages of ‘abababababab’, do we compensate them with the same number of bananas? If Willy sees that another monkey got paid the same bananas for writing ‘abababab’ he would be less inclined to write another work of a similar stature. Without the incentive to produce works like *Hamlet*, Willy may never have gone on to type *Macbeth*, *Othello*, *King Lear*, or *The Tempest.* In order to promote the writing of the very best literature, it is essential to establish a method to determine how to reward each monkey based on the quality of their work.

The mathematics of Kolmogorov complexity combines classical probability theory and information theory to provide a means of assessing the algorithmic information content of a sequence of characters. The Kolmogorov complexity K of an object $x$ is the length of the shortest binary program that outputs the object, which is a measure of the absolute information of the individual object.

We can see right away that a very short program of only two lines of code will output the second monkey’s sequence of ‘ababab’:

```
for i = 1 to 25
print ab
```

Rather than use the fictitious monkey that types ‘ababab...’, in order to establish a useful comparison we will pit Willy against Jimmy, another famous monkey from the typing pool. By considering the reception of literature through a comparative analysis, we have expanded the typing monkey scenario to include the cultural phenomenon that closely followed the first production of literature: the literary competition.

*Hamlet* and *Finnegans Wake* have been submitted by Willy and Jimmy respectively, and we must determine to whom we will award the laurels. Since monkeys are little interested in actual laurels, our literary competition will award the Top Banana prize.

In his book *Joyce, Chaos and Complexity*, Thomas Jackson Rice analysed the algorithmic complexity of *Finnegans Wake*, and drew the following conclusion:

"Since [Jimmy's] initial algorithms are now 'givens', available in his manuscript material and elsewhere, the algorithmic complexity of *Finnegans Wake* is in the low to moderate range." [2] We will see how it compares to the algorithmic complexity of *Hamlet*.

Unfortunately for us, Kolmogorov proved in his Noncomputability Theorem that K is uncomputable [3]; however, we are not completely at a loss. Solomonoff, another major contributor to the theory of Kolmogorov complexity, said of K that "it is clear that many of the individual terms of Eq. (1) are not 'effectively computable' in the sense of Turing [... but can be used] as the heuristic basis of various approximations." [4]

Shannon's stochastic entropy H provides a useful approximation to K. Classical information theory holds that a random variable $X$ distributed according to $P(X = x)$ has entropy

$$H(X) = -\sum P(X = x) \log P(X = x)$$

where the interpretation is that $H(X)$ bits are on the average sufficient to describe an outcome $x$. Kolmogorov proved that stochastic entropy and expected algorithmic complexity are equal [3], so we can calculate $H(Hamlet)$ and $H(FW)$ to approximate K. We will calculate values to compare the entropy of the distributions of words, sentences, and the entire texts of *Hamlet* and *Finnegans Wake*.

The source text used for *Hamlet* was taken from the first folio edition, available at Project Gutenberg [5]. The text was stripped of all punctuation marks so it consisted of the 26 letters plus spaces, which resulted in a text of 138902 characters. The text of *Finnegans Wake* was taken from the plain text version available from Finnegans Web [6], stripped similarly to *Hamlet*, and truncated as $FW'$ to the first 138902 characters so the text will be the same length as *Hamlet*. The entropy calculations for words and sentences were performed using SRILM (Stanford Research Institute Language Modelling Toolkit) [7]. We first calculated the stochastic entropy of the distribution of words in *Hamlet* and in $FW'$:

$$H_{words}(Hamlet) = 9.4 \;\; H_{words}(FW') = 10.4$$

So, we have

$$H_{words}(Hamlet) < H_{words}(FW')$$

This means that on average, a text that contains all the words in *Hamlet* will be less complex than a text that contains all the words in $FW'$. This meets with our expectations because there are about half as many unique words in *Hamlet* as there are in $FW'$. However, it would be premature to conclude that *Hamlet* is less complex than *Finnegans Wake*, because the stochastic entropy of the words does not take into account their syntactic relations with each other.

We need a complexity measure that takes into account the discernible patterns in the way Willy and Jimmy combined the words in *Hamlet* and *Finnegans*

*Wake.* We can turn to n-gram language models commonly used in Natural Language Processing to compare the entropy of sentences in *Finnegans Wake* and *Hamlet.* The n-gram model will test how much the current word of a sentence depends on a constant number of preceding words in the sentence. The entropy H of a sentence $S$ consisting of words $w_1w_2w_3 \dots w_m$ calculated according to an n-gram model is then

$$H_n(S) = \log P(w_1)P(w_2|w_1)P(w_3|w_2w_1)\prod_{i=4}^{n} P(w_i|w_{i-1}\dots w_1)$$

A trigram model was used to compute the average entropy of the distribution of sentences in *Hamlet* and *FW'*:

$$\bar{H}_3(Hamlet) = 4.8 \;\; \bar{H}_3(FW') = 6.2$$

So, we have

$$\bar{H}_{sentence}(Hamlet) < \bar{H}_{sentence}(FW')$$

The inequality holds for the widely used trigram model. So, we know that the entropy of *Hamlet* is less than that of *Finnegans Wake* based on lexical effects (which words are used), and non-lexical effects (how the words are combined into sentences).

In his paper 'The Complexity and Entropy of Literary Styles' Kontoyiannis has suggested that the Markov model that is implicit in n-gram language models gives a poor estimate of the entropy of an entire text:

> It seems to be a well-understood fact that, as already argued by Chomsky 40 years ago, Markovian models are not adequate linguistic descriptions for natural languages. From our point of view (that of entropy estimation), one obvious deficiency of Markov models is that they have [parametric] bounded context-depths and thus cannot capture the long-range dependencies encountered in written English [8].

Kontoyiannis describes a method for entropy estimation of an entire text via string matching, which is related to the Lempel-Ziv compression algorithm [8]:

We model text as a string produced by a stationary process $X = \{\dots, X_{-1}, X_0, X_1, X_2, \dots\}$, with each $X_i$ taking values in a finite alphabet A (like the 26 letters). Suppose we are given a long realization of this process (like *Hamlet* or *Finnegan's Wake*) starting at time zero: $x_0x_1 \dots x_M$. For each position $i \geq 1$ of the "text" $x_0x_1 \dots x_M$ we calculate the length of the shortest prefix starting at $x_i$, that does not appear starting anywhere in the previous $i$ symbols $x_0x_1 \dots x_{i-1}$, and denote this length by $l_i$. (We allow the possibility that there is overlap between the prefix starting at $x_i$ and the matching string starting somewhere in $x_0x_1 \dots x_{i-1}$). The entropy estimator is given by the formula:

$$\hat{H}_N = \left[\frac{1}{N}\sum_{i=1}^{N}\frac{l_i}{\log(i+1)}\right]^{-1}$$

for some $N < M$. As $i$ grows, there is no restriction on how far back into the past we can look for a match [8]. The entropy estimation takes into account all of the text that has appeared previously, which gives a better estimate of the entropy of the entire text. The values for the two texts are as follows:

$$H_{text}(Hamlet) = 2.08 \ \ H_{text}(FW') = 2.44$$

So, we have

$$H_{text}(Hamlet) < H_{text}(FW')$$

Our entropy estimators for the distributions of words, sentences, and the entire texts in *Hamlet* and *Finnegans Wake* gave the same inequality, so we can conclude in our estimate

$$K(Hamlet) < K(FW')$$

and award the Top Banana prize to Jimmy the monkey for producing the work with greater complexity.

Although Rice claimed the algorithmic complexity of *Finnegans Wake* is in the low to moderate range, we must conclude that it can only have low to moderate complexity if *Hamlet* is to be considered as having extremely low complexity, which is not a satisfactory conclusion.

Strictly speaking, K measures the absolute algorithmic information content of an object. We have used three approximations to K, and for each of the approximations the inequality holds. Because K is noncomputable, we cannot draw any provable conclusions about K; however, we can use the approximations to determine the outcome of the literary competition with a great deal of confidence because, as we all know, literary criticism has never been an exact science.

## References

[1] ReMine, W. J. The Biotic Message: Evolution versus Message Theory. *St. Paul Science*: Saint Paul, 1993.

[2] Rice, Thomas Jackson. *Joyce, Chaos, and Complexity.* University of Illinois Press: Urbana, 1997.

[3] Attributed to A. N. Kolmogorov in A. K. Zvonkin and L.A. Levin. The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms. *Russian Math. Surveys*, 25(6):83-124, 1970.

[4] Solomonov, R. J. A formal theory of inductive inference, part 1 and part 2. *Inform. Contr.*, 7:1-22, 224-54, 1964.

[5] Shakespeare, William. *Hamlet.* Project Gutenberg. http://www.gutenberg.org/

[6] Joyce, James. *Finnegans Wake.* Finnegans Web. http://www.trentu.ca/jjoyce/fw.htm

[7] Stanford Research Institute Language Modeling Toolkit http://www.speech.sri.com/projects/srilm/

[8] Kontoyiannis, I. The Complexity and Entropy of Literary Styles. NSF Technical Report No. 97, Department of Statistics, Stanford University, 1997.

[9] Ziv, J, and A. Lempel. Compresion of Individual Sequences by Variable Rate Encoding. *IEEE. Trans. Inf. Theory.* 24(5):530-536, 1978.

Thanks to Dr. Jeffrey Shallit, my former formal languages professor, for providing comments to help ensure the mathematical validity of the argument. Any errors, typos or omissions are my own.

This paper is typeset in LaTeX, the standard format for mathematics publishing based on algorithms designed by Dr. Donald E Knuth, author of *The Art of Computer Programming.*

Published June 28, 2004, in honour of Dr. Christiam oBok's departure from Toronto to assume his position as Projector of the Academy.