18

Crosswords and Codebreaking

In this chapter we make a random walk through a few topics related to language modelling.

▶ 18.1 Crosswords

The rules of crossword-making may be thought of as defining a constrained channel. The fact that *many* valid crosswords can be made demonstrates that this constrained channel has a capacity greater than zero.

There are two archetypal crossword formats. In a 'type A' (or American) crossword, every row and column consists of a succession of words of length 2 or more separated by one or more spaces. In a 'type B' (or British) crossword, each row and column consists of a mixture of words and single characters, separated by one or more spaces, and every character lies in at least one word (horizontal or vertical). Whereas in a type A crossword every letter lies in a horizontal word *and* a vertical word, in a typical type B crossword only about half of the letters do so; the other half lie in one word only.

Type A crosswords are harder to *create* than type B because of the constraint that no single characters are permitted. Type B crosswords are generally harder to *solve* because there are fewer constraints per character.

Figure 18.1. Crosswords of types A (American) and B (British).

Why are crosswords possible?

If a language has no redundancy, then any letters written on a grid form a valid crossword. In a language with high redundancy, on the other hand, it is hard to make crosswords (except perhaps a small number of trivial ones). The possibility of making crosswords in a language thus demonstrates a *bound on the redundancy* of that language. Crosswords are not normally written in genuine English. They are written in 'word-English', the language consisting of strings of words from a dictionary, separated by spaces.

▷ Exercise 18.1.[2] Estimate the capacity of word-English, in bits per character. [Hint: think of word-English as defining a constrained channel (Chapter 17) and see exercise 6.18 (p.125).]

The fact that many crosswords can be made leads to a lower bound on the entropy of word-English.

For simplicity, we now model word-English by Wenglish, the language introduced in section 4.1 which consists of W words all of length L. The entropy of such a language, per character, including inter-word spaces, is:

$$H_W \equiv \frac{\log_2 W}{L+1}. \tag{18.1}$$

We'll find that the conclusions we come to depend on the value of H_W and are not terribly sensitive to the value of L. Consider a large crossword of size S squares in area. Let the number of words be $f_w S$ and let the number of letter-occupied squares be $f_1 S$. For typical crosswords of types A and B made of words of length L, the two fractions f_w and f_1 have roughly the values in table 18.2.

	A	B
f_w	$\frac{2}{L+1}$	$\frac{1}{L+1}$
f_1	$\frac{L}{L+1}$	$\frac{3}{4}\frac{L}{L+1}$

Table 18.2. Factors f_w and f_1 by which the number of words and number of letter-squares respectively are smaller than the total number of squares.

We now estimate how many crosswords there are of size S using our simple model of Wenglish. We assume that Wenglish is created at random by generating W strings from a monogram (i.e., memoryless) source with entropy H_0. If, for example, the source used all $A = 26$ characters with equal probability then $H_0 = \log_2 A = 4.7$ bits. If instead we use Chapter 2's distribution then the entropy is 4.2. The redundancy of Wenglish stems from two sources: it tends to use some letters more than others; and there are only W words in the dictionary.

Let's now count how many crosswords there are by imagining filling in the squares of a crossword at random using the same distribution that produced the Wenglish dictionary and evaluating the probability that this random scribbling produces valid words in all rows and columns. The total number of *typical* fillings-in of the $f_1 S$ squares in the crossword that can be made is

$$|T| = 2^{f_1 S H_0}. \tag{18.2}$$

The probability that one word of length L is validly filled in is

$$\beta = \frac{W}{2^{L H_0}}, \tag{18.3}$$

and the probability that the whole crossword, made of $f_w S$ words, is validly filled in by a single typical in-filling is

$$\beta^{f_w S}. \tag{18.4}$$

So the log of the number of valid crosswords of size S is estimated to be

$$\log \beta^{f_w S} |T| = S\left[(f_l - f_w L) H_0 + f_w \log W\right] \log \beta^{f_w S} |T| \tag{18.5}$$

$$= S\left[(f_l - f_w L) H_0 + f_w (L+1) H_w\right] \tag{18.6}$$

which is an increasing function of S only if

$$(f_1 - f_w L) H_0 + f_w (L+1) H_w > 0. \tag{18.7}$$

So arbitrarily many crosswords can be made only if there's enough words in the Wenglish dictionary that

$$H_W > \frac{(f_w L - f_1)}{f_w (L+1)} H_0. \tag{18.8}$$

Plugging in the values of f_1 and f_w from previous page, we find the following.

Crossword type	A	B
Condition for crosswords	$H_W > \frac{1}{2}\frac{L}{L+1} H_0$	$H_W > \frac{1}{4}\frac{L}{L+1} H_0$

If we set $H_0 = 4.2$ bits and assume there are $W = 4000$ words in a normal English-speaker's dictionary, all with length $L = 5$, then we find that the condition for crosswords of type B is satisfied, but the condition for crosswords of type A is *only just* satisfied. This fits with my experience that crosswords of type A usually contain more obscure words.

Further reading

These observations about crosswords were first made by Shannon (1948); I learned about them from Wolf and Siegel (1998). The topic is closely related to the capacity of two-dimensional constrained channels. An example of a two-dimensional constrained channel is a two-dimensional bar-code, as seen on parcels.

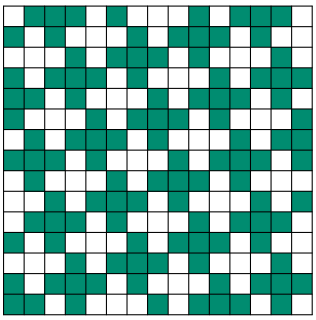

Figure 18.3. A binary pattern in which every pixel is adjacent to four black and four white pixels.

Exercise 18.2.[3] A two-dimensional channel is defined by the constraint that, of the eight neighbours of every interior pixel in an $N \times N$ rectangular grid, four must be black and four white. (The counts of black and white pixels around boundary pixels are not constrained.) A binary pattern satisfying this constraint is shown in figure 18.3. What is the capacity of this channel, in bits per pixel, for large N?

▶ 18.2 Simple language models

The Zipf–Mandelbrot distribution

The crudest model for a language is the monogram model, which asserts that each successive word is drawn independently from a distribution over words. What is the nature of this distribution over words?

Zipf's law (Zipf, 1949) asserts that the probability of the rth most probable word in a language is approximately

$$P(r) = \frac{\kappa}{r^{\alpha}}, \tag{18.9}$$

where the exponent α has a value close to 1, and κ is a constant. According to Zipf, a log–log plot of frequency versus word-rank should show a straight line with slope $-\alpha$.

Mandelbrot's (1982) modification of Zipf's law introduces a third parameter v, asserting that the probabilities are given by

$$P(r) = \frac{\kappa}{(r+v)^{\alpha}}. \tag{18.10}$$

For some documents, such as Jane Austen's *Emma*, the Zipf–Mandelbrot distribution fits well – figure 18.4.

Other documents give distributions that are not so well fitted by a Zipf–Mandelbrot distribution. Figure 18.5 shows a plot of frequency versus rank for the LaTeX source of this book. Qualitatively, the graph is similar to a straight line, but a curve is noticeable. To be fair, this source file is not written in pure English – it is a mix of English, maths symbols such as 'x', and LaTeX commands.

Figure 18.4. Fit of the Zipf–Mandelbrot distribution (18.10) (curve) to the empirical frequencies of words in Jane Austen's *Emma* (dots). The fitted parameters are $\kappa = 0.56$; $v = 8.0$; $\alpha = 1.26$.