The maths of language

By Freya Jackson

People often ask me, ‘Freya, what A-levels are you studying?’

And I tell them, ‘I’m doing Maths, Further Maths, Chemistry…’ Now at this point they think they have me all figured out, and they’ve already put me in the STEM box. They think they can take a good guess at what comes next, so then I hit them with it: ‘…and French’. Boom. Now they’re confused. Why could she be possibly studying a language along with those subjects? What could they possibly have in common?

Okay perhaps that was a little too dramatic. But you get the point – maths and languages are usually considered to be on opposite points of the academic spectrum. But in this essay I aim to show you how they are actually more linked than you probably thought (and maybe redeem my A-level choices along the way). I hope you enjoy this foray into the fascinating relationship between words and numbers.

The infinite dictionary

All languages are built from a finite number of building blocks. English has around 40 phonemes, which are the smallest units of sound that we can use to distinguish one word from another. Examples of phonemes include ‘b’, ‘oo’, ‘ch’ and ‘ng’. On their own, they usually mean nothing. But combine them, and suddenly we have words. Combine words, and we get sentences. Combine sentences, and we get essays written the night before a deadline…

Using this small set of phonemes we can create potentially infinite combinations – just like how the digits 0 to 9 can generate infinitely many numbers. Mathematicians call this Combinatorics. Say we wanted to calculate the number of combinations we could make from these 40 phonemes, but only using each phoneme once. We could write that as 40! – but including combinations of length 1 to 40, the equation would be:

$$S = \sum_{k=1}^{40} \frac{40!}{(40-k)!}$$

Now this gives us roughly 8.16 x 10^47 possibilities, but luckily for us not every phoneme combination gives us a meaningful word. The current Oxford English Dictionary holds around 520,000 entries and the average English speaker knows about 25,000 words. This is still an impressive number of combinations to memorise, especially since phoneme combinations are so random.

Or are they?

Phonotactics

Here’s where we start applying a set of linguistic rules called Phonotactics. Every language has its own Phonotactics, but for ease of explanation I’m going to go through some of the ones for English.

Firstly, English forbids certain phoneme combinations simply because they’re hard to pronounce. Don’t believe me? Try saying ‘onghaebithuzhareplu’. It also favours certain combinations that give easily heard contrasts so listeners can tell sequences apart, and combinations can be created or removed over time due to historical change (such as vowel or consonant shifts) and borrowing from other languages.

Grammar

Everyone's favourite part of learning a language.

Grammar is actually quite mathematical, as it can be reduced to a system of rules and structures. Take the example of conjugating French verbs: for each subject there is a specific ending depending on the tense you are using. This system is common to many languages such as Arabic, where the subject specific affix goes onto the front of verbs instead.

Can we model French verb conjugation mathematically? We can try.

Let a verb be represented as:

$$V = (S, I)$$

Where S= stem and I= Infinitive verb ending (1st, 2nd, 3rd → er, ir, re).

Define conjugation as a function:

$$C(V, t, p) = S'(t, p, I) + E(I, t, p)$$

Where t= tense, p= person (1–6), $S'(t, p, I)$= possibly modified stem and $E(I, t, p)$= ending.

Now let's try this with the verb *Parler* (to speak):

Let $I = 1$ (first group, er verbs) and $S =$ 'parl'

Then:

$$E(1,\text{present}, p) = \begin{cases} e & p = 1 \\ es & p = 2 \\ e & p = 3 \\ ons & p = 4 \\ ez & p = 5 \\ ent & p = 6 \end{cases}$$

This model can be replicated for most verbs in the French language, though there are a few irregulars – but then what good mathematical rule doesn't have exceptions?

Recursion

When the linguist Noam Chomsky set out his theory for Universal Grammar (the theory that all languages in the world share some common features in grammar), one of its key aspects was recursion. This is the idea that a structure can be embedded within itself. Humour me for a moment and consider the nursery rhyme:

There was an old lady who swallowed a cow
I don't know how she swallowed a cow.
She swallowed the cow to catch the dog
She swallowed the dog to catch the cat
She swallowed the cat to catch the bird
She swallowed the bird to catch the spider
She swallowed the spider to catch the fly.

This is an example of recursion – we could imagine this going on forever. We can also represent it mathematically by defining S(n) as the object swallowed at step n (so S(1)=fly, S(2)=spider, …). Let R(n) be the nested expression representing all swallows up to n. Then:

R(1) = S(1)

R(n) = S(n) ° R(n-1) for n ≥ 2

Where ° denotes ‘swallowed to catch’. Expanding, we get

R(n) = S(n) ° S(n-1) ° … ° S(1).

Now we’ve successfully just expressed a nursery rhyme mathematically!

Predicting what comes next

If I say ‘I would like a cup of…’, you will probably be thinking of either ‘tea’ or ‘coffee’ rather than ‘maths’.

Language is surprisingly predictable, and this predictability can be modelled using probabilistic frameworks such as Markov chains, where the probability of a word depends on the preceding context. It can also be modelled using graph theory. Words become nodes, and relationships between them become edges. Clusters form around related concepts, and highly connected nodes correspond to frequently used or semantically rich words.

We can do a similar thing with languages themselves. Here is a graph showing lexical differences among different European languages:

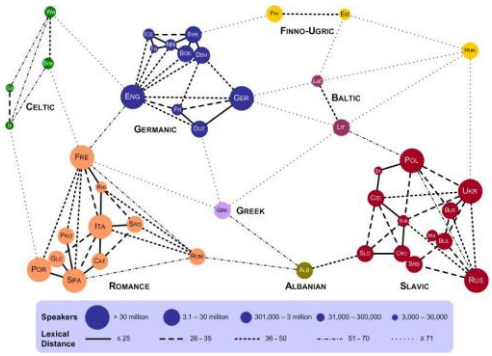

Figure 1: Lexical differences among European languages [1]

We can see that the clusters relate to different language families. Graphing is a good way to visualise linguistic relationships, but it’s not the only way to find relationships between languages.

Computational linguistics

Computational linguistics has many useful applications, among them finding language relationships by comparing words and sounds with simple similarity measures and by grouping languages that look most alike. The same techniques help record under-researched languages quickly and build basic speech or translation tools.

For the under-documented Varli dialect spoken in Western India, researchers collected wordlists and matched similar words to nearby Konkani and Marathi languages to find likely relatives, then reused models trained on Marathi to build a starter speech recogniser and tiny translator. This was all done using statistics, which really shows the applicability of such tools to processing language.

Zipf’s law

This brings us onto one of the most striking mathematical patterns in language! Zipf’s law states that frequency of a word in a language is inversely proportional to its rank in a frequency table.

This means the most common word in a language appears roughly twice as often as the second most common, three times as often as the third, and so on. When plotted on a log-log graph, this produces a near-linear relationship.

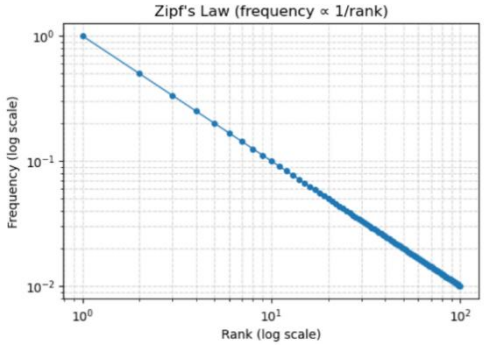

Figure 2: graph showing Zipf's law on a log-log graph [2]

This pattern appears across all languages – from English to Arabic and even Varli. The universality of Zipf's law suggests that it reflects a deeper principle. One interpretation is that it represents a balance between speaker effort and listener comprehension. A small set of highly frequent words allows efficient communication, while a large set of rarer words provides precision when needed. This shows similarity with maths, as we see operations such as '=', '+' and 'x' come up more frequently than others.

So does maths follow Zipf's law too? I tried to find out by analysing the number of times each mathematical symbol came up across 8 different maths textbooks covering a range of topics, and plotting them in a graph:

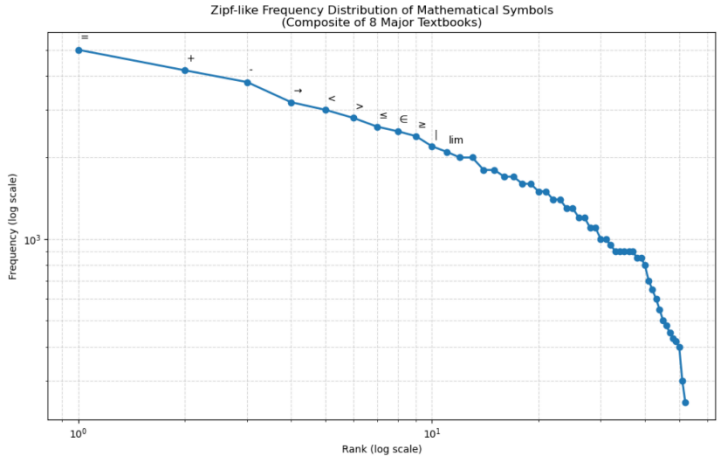

Figure 3: graph showing Zipf-like distribution of mathematical symbols from 8 textbooks [2]

It's not perfect, but it does show a very clear trend – and it seems to follow Zipf's law! A group of researchers at Oxford University did something similar using the Feynman lectures of physics, named equations on Wikepedia and Cosmic Inflation equations, and here are their results:

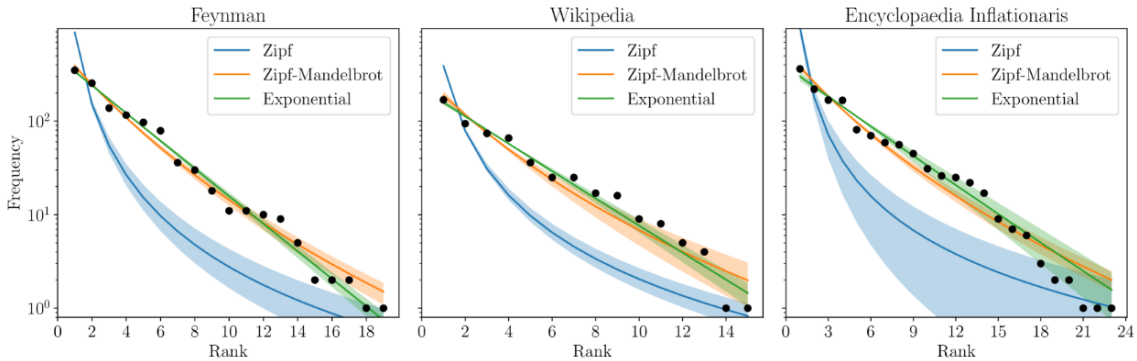

Figure 4: graphs showing distribution of mathematical symbols from different sources [3]

Their results (which may be a little more reliable than mine) also show a strong similarity to Zipf's law. Interestingly, when the researchers applied the same analysis to randomly generated equations, the Zipf-like pattern disappeared. Could this reveal how human brains approach problem-solving? Or is it a reflection of an inherent structure in the universe itself, present in both languages and maths? The answers could be crucial in finding out more about the basic principles that govern our world.

Not only have we talked about the maths of language, we have now analysed the language of maths!

Too many graphs

There are many linguistic features that can be graphed such as speech rate, word order frequencies, discourse marker frequency, loanword percentage and so many more. The fact that so many linguistic features can be graphed suggests that language has an underlying statistical geometry.

One of the graphable features is information per syllable. What's interesting about this is the equation used to represent the average amount of information (or unpredictability) per linguistic unit:

$$H = -\sum p(x) \log\ p(x)$$

Look familiar? This is the formula for entropy, used many scientific applications such as thermodynamics. But in 1948, the mathematician Claude Shannon used this equation to model linguistics! Here's how it works:

Let X be a random variable representing a linguistic unit (e.g. a word), with probability distribution $p(x)$.

We can interpret this as:

$$H(X) = \mathbb{E}[-\log\ p(x)]$$

Where $-\log p(x)$ = information carried by word x and $H(X)$ = average information per word, or the average information (unpredictability) per word in the language.

We can test this using extremes - if one word dominates then:

$$H \to 0$$

And if all words are equally likely then:

$$H = \log\ |\ X\ |$$

I hope that wasn't too much information per syllable!

How language shaped maths

Some languages use a base-10 counting system. Some languages use base-20. And some languages use base-60...

The Babylonian language had a base-60 counting system, but this clearly was not chosen for mathematical elegance. It actually came about from a linguistic merging of two earlier counting systems: a base-10 system from Sumerian and a base-6 system from Akkadian. Naturally, the new hybrid counting system grouped numbers in sixties.

This gave the Babylonians far more divisibility than we have today; 60 can be expressed as $2^2 \times 3 \times 5$ while 10 only as 5 x 2. As a result, in ancient Babylonia fractions were easier to express, reciprocals were often finite and tables of squares and reciprocals were compact and efficient. The Babylonians were prolific

mathematicians and are accredited with developing sophisticated algebraic methods, astronomical calculations and early trigonometric ideas.

However their most enduring legacy is arguably the fact that we still count time in base-60. There are 60 seconds in a minute, and 60 minutes in an hour not because these are 'natural' mathematical facts, but because of our linguistic heritage.

Conclusion

So, are maths and languages really opposites?

Not quite.

Languages are built from discrete units, structured by rules, shaped by probability, and optimised for efficiency. They can be modelled with graphs, analysed with statistics, and even coded on a computer.

Which means that perhaps my A-level choices weren't so contradictory after all.

Or, at the very least, I now have a mathematical justification for procrastinating in two languages instead of one.

Bibliography

[1] Tishchenko, K. (1997) recreated by Elms, T. Sourced from https://blogs.cornell.edu/info2040/2018/09/17/38095/

[2] Graphs made by me using Matplotlib

https://matplotlib.org/

The textbooks I used for my second graph were: *Calculus* (Stewart, S.), *Calculus* (Spivak, M.), *Introduction to Linear Algebra* (Strang, G.), *Linear Algebra and Its Applications* (Lay, D.), *Elementary Differential Equations* (Boyce, DiPrima), *Differential Equations and Dynamical Systems* (Hubbard, West), *Probability and Random Variables* (Grimmett, Stirzaker) and *Introduction to Probability* (Bertsekas, Tsitsiklis).

[3] Constantin, A., Bartlett, D., Desmond H., Ferreira, P. (2024) *Statistical Patterns in the Equations of Physics and the Emergence of a Meta-Law of Nature* Sourced from

https://arxiv.org/html/2408.11065v1?utm_source=copilot.com