Language Learning Diary - Part Six

Linas Vepštas

Feb 2022 - March 2022

Abstract

The language-learning effort involves research and software development to implement the ideas concerning unsupervised learning of grammar, syntax and semantics from corpora. This document contains supplementary notes and a loosely-organized semi-chronological diary of results. The notes here might not always makes sense; they are a short-hand for my own benefit, rather than aimed at you, dear reader!

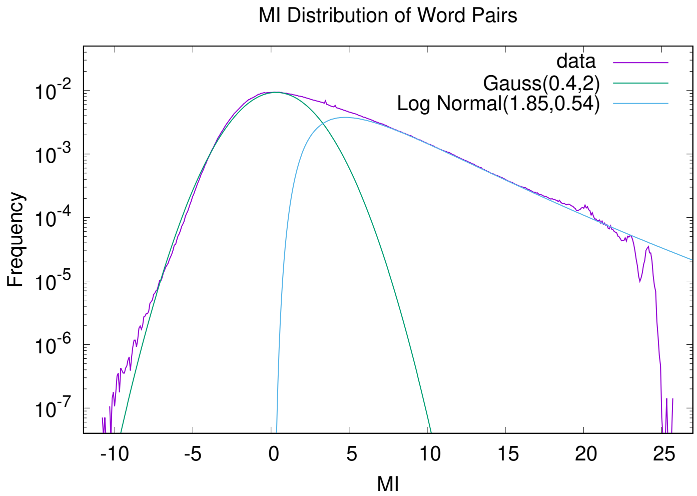
Introduction

Part Six of the diary on the language-learning effort re-examines some older data from a physical-science, graph-theoretic, information-theoretic viewpoint. The “older data” here is primarily the data for word-pairs. The first section defines the “density of states” in terms of the “energy”. These are fundamental physics concepts appearing in thermodynamics; they’re mapped onto the analogous word-pair statistics concepts. The second section examines a number of other quantities in this same conceptual framework. Some of this repeats results reported earlier; here, a more comprehensive approach is taken.

There is an implicit meta-goal, which is not achieved: to provide a statistical-mechanical, field-theoretic framework for the language data. This is explored only at a rather superficial level. It feels like deeper analogies are certainly possible, but it is not clear how these could offer insight.

Summary Conclusions

The most important result presented here is an analysis of the word-pair MI distribution. For the first time, it becomes clear that it factors into two parts: a Gaussian distribution, arising from randomly-paired words, and a log-normal distribution, arising from word-pairs that carry actual syntactic information. This is obvious, in retrospect, and was always visible; just that now, we have an rough explanation for the shape. Here’s the relevant graph, in full glory; it is explained towards the end of the chapter.



This chapter starts with definitions of some abstract concepts, followed by some data analysis. In order of appearance (and not in order of importance):

- The product topology. The space of natural-language sentences is simply the collection of ordered strings of words. A “sentence” is just the Cartesian product of words in a vocabulary. As a Cartesian-product space, it has a natural “topology”, the “product topology”. The basis of product topologies are called “cylinder sets”; these are just sequences of specific words, interspersed with wild-cards. Word-pairs are just specific cylinder sets: two words, with zero or more wild-cards between them, and an arbitrary number of wild-cards before and after them.

- Density of states, theoretical definition. A “state” can be identified with a cylinder set in the product topology. The “energy” of a state can be identified with the log of the probability of that state (equivalently, the log of the measure of the cylinder set). The density of states is then simply the distribution of the states, as a function of their energy. That is, for a given small but fixed interval of energy, how many states are there in that interval? This is the density of states. In thermodynamics and chemistry, this is a fundamental concept; for natural language, it is novel, but worth asking about to see if any analogies hold.[1]

- Density of states, experimental result. Consider the collection of all observed word-pairs (w_j, w_k). The frequency with which some word-pair is observed is $p(w_j, w_k)$ and the energy is $E = -\log_2 p$. The density of states $\rho(E)$ is then just a histogram: how many word-pairs were observed in a small, finite-sized

[1] For example, in chemistry, there are lots of low-energy states at low temperatures; it is hard to have many high-energy states at low temperatures. Typical distributions are the Maxwell-Boltzmann distribution for an ideal gas; the Fermi-Dirac distribution for fermions, and the Bose-Einstein distribution for supercooled quantum states. The first is conventionally taught in college chemistry. Is there anything analogous in natural language?

interval of energy? Making this histogram, one easily finds that, to first order, its a nice straight line (on a semi-log graph), so that the density of states is $\rho(E) \sim 2^{-E}$ over a wide range, dropping off at the low and the high end due to under-sampling effects. This is, more or less, with some twists, a rephrasing of the old and well-known result that the Zipfian distribution applies to word-pairs.

- The Zipf graph goes very nearly as 1/rank i.e. as the classical Zipf with exponent 1. However, it has a bit of a hump, as does $\rho(E)$. Looking more closely, at the top-1200 ranked pairs, the Zipf exponent is 3/4 (almost exactly) and not 1. This is a so-called "small-world" exponent. The open-world exponent of 1 kicks in above 1200. This suggests that all word-pairs above 1200 are under-sampled. This is out of a total of 10^7distinct word-pairs that were observed. The small world is indeed small, the provinces vast.
- The $\rho(E)$ has a similar hump. The constant slope can be removed by rescaling to $2^E\rho(E)$ which revels the precise form of the hump: it is exactly a (log-normal) Gaussian!
- Closer examination reveals that the idea of a measure on a product topology is naive and incorrect. The first problem is that the size of the vocabulary is not fixed; the larger the corpus, the more new words are found (proper names, geographical place-names, slang, marketing terms, technical terms...) In the limit of infinite vocabulary, this would imply that the measure is log-divergent, i.e. is not a measure.
- A better theoretical foundation is needed. (None is proposed here) Such a foundation would need to explain and characterize:
  - The under-sampling effect, and the location of the large-world to small-world cross-over.
  - The effect of human-scale finite sentence lengths on punctuation and determiners.
- The under-sampling effect is foundational, and affects the graphed distributions in all graphs in all chapters of this diary. It's pervasive, and confounding, and makes it difficult to understand "what's actually happening". A preliminary sketch is made for how to untangle sample-size effects is given.
- Earlier chapters explored marginal probabilities, fractional entropies, mutual information, marginal MI and so on. These are re-examined again here, this time as functions of E
- Vertex-degree graphs are presented. Vertex-degree graphs are commonplace in network analysis; it seems fitting to do that here. A vertex is a word, and it's degree is the number of (distinct, unique) word-pairs it occurs in. For the range of $10 \lesssim D \lesssim 1200$, the probability $p(D)$ of observing a word-vertex with degree D goes as $p(D) \sim D^{-1.6}$. This is a small-world scaling exponent; it is far away from being a scale-free network exponent. Note that this is a statement about infrequent words; common words, like "the" will have a degree in the millions.

- The word-pair MI distribution is composed of two parts. One part is a Gaussian, centered more or less on an MI of zero. This Gaussian is purely due to selections of word-pairs having no syntactic relationships, and contains no syntactic information. Subtracting this leaves behind the word-pairs with the actual syntactic information. That distribution seems to be log-normal, i.e has strictly-positive MI.
- Distributions of the word-disjunct MI are presented. They vaguely resemble the word-pair MI graphs, but are dirtier/uglier. No particular insight is gained.
- The ranked-MI looks vaguely like a Laplacian. Two ideas are developed: a fibered-Laplacian, and a Hamming-Laplacian. Experimental data is shown for the Hamming-Laplacian. It's curious, but provides no particular insight.

That's it. Now on with the main text.

Field Theory Models and Statistical Mechanics

Field theory models applied to statistics and language have surely been thrashed to death in the literature (of which I am only dimly aware; thus, no bibliography.). The below is an attempted recap of some basic ideas, recast into the notation used locally in this diary. After a few basic initial definitions, it rapidly devolves into a presentation of experimental results (for word-pairs).

Density of States

Starting point is a discrete linear lattice of words in a sentence. Associated to each sentence is a probability $p(w_1, \cdots, w_n)$ for words w_k and a sentence of length n. We do not know that probability; we just assume it exists *a priori*. We can make experimental pair-wise observations of word-pairs as $(*, *, \cdots, *, w_i, *, \cdots, *, w_j, *, \cdots, *)$ of pairs of words (w_i, w_j) within the full sentence. Note the former is a cylinder set, *i.e.* an element of the product topology on strings.

Let $\sigma = (w_1, \cdots, w_n)$ be a string (the sentence). Define the energy of a string as $E(\sigma) = -\log_2 p(\sigma)$ and define the energy density as

$$\begin{aligned}\rho(E) &= \sum_{\sigma} \delta(E - E(\sigma)) \\ &= C \sum_{w_i, w_j} \delta(E + \log_2 p(w_i, w_j))\end{aligned}$$

where $\delta(x)$ is the Dirac delta function (in principle) or just a finite–width, but thin Gaussian in practice, or, more plainly, just a box filter, so that we can do histogram counting. The constant C appears because the sum over pairs is a multiple of the sum over all states; it over-counts (since $\sum_{\sigma} = \sum_{w_1, w_2, \cdots, w_n}$ counts all words at all word-positions.) A formal derivation of the value of C from first principles seems tedious and unenlightening. Not to worry, we can force it experimentally simply by requiring that

$$\int \rho(E)\, dE = 1$$

I honestly do not recall if any of the prior diary entries ever supplied a graph of $\rho(E)$. Better late than never?

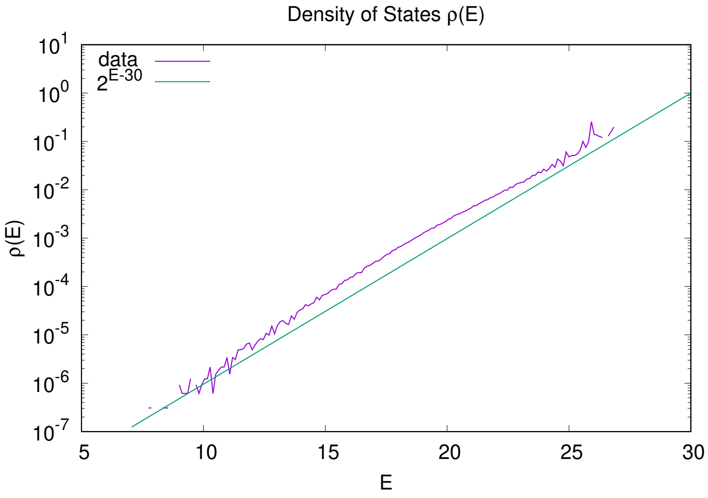
Interpretation

The above definition of the density of states is motivated by the Boltzmann distribution $p = e^{-\beta E}$. Taking the log of both sides and setting $\beta = 1$ gives $E = -\log p$.[2]

Density of States - Experimental Result

Working with the Run-1 dataset ‘run-1-en_pairs-tranche-123.rdb‘. This dataset is characterized in the subsection below. To generate the histogram, simply create N bins, and increment by one whenever $-\log_2 p(w_i, w_j)$ lies within the bin boundaries.[3]

The graph below uses 200 bins, running between a lower bound of 7.0 and upper bound of 30.0. Thus, the width of each bin is $dh = 23/200$. The data is as marked, and, to provide a sense of scale, the line 2^{E-30} is graphed. Note that there is a scattering of dots at the upper-right and lower left (zoom in to see them). Dots correspond to non-empty bins in the histogram, with empty neighbors. These dots have a special significance.



This graph can be understood as a kind-of upside-down Zipfian distribution. The scatter of dots at the top-right are the pairs that were seen only a handful of times. The topmost, rightmost dot corresponds to the word-pairs that were observed only once, and

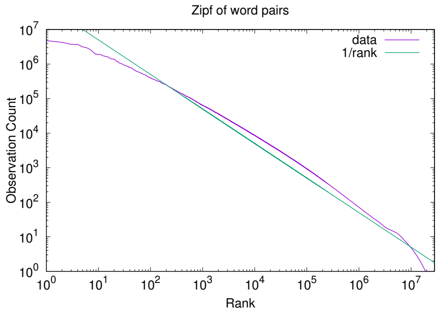
[2] See Wikipedia, Boltzmann distribution for additional details.

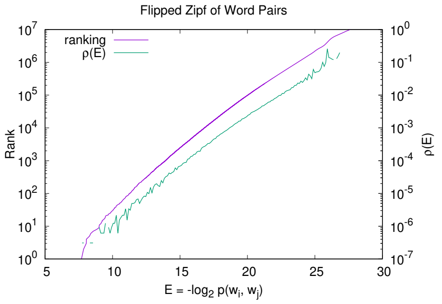
[3] Use the script in ‘utils/density-of-states.scm‘

thus have a very high E. Specifically, $E_1 = \log_2 985483375 \approx 29.8763$ for this point, as there was a grand total of 985 million pairs observed. The density here is a Dirac delta spike, since there were 9215082 distinct, unique word-pairs observed exactly once; thus $\rho(E_1)$ is normalized to $985483375/9215082/dh$. The next dots correspond to the number of distinct word-pairs that were observed only twice, then three times, *etc.* until they run together into common bins in the histogram. The dots at the bottom-left correspond to word-pairs there are extremely common (typically involving the words “the”, “a”, punctuation.) These would be ranked first in a Zipfian distribution, thus the bottom-left of this graph corresponds to the top-left of a Zipf graph.)

Note that if the counts in the right-most bins are smeared, so that they are not delta functions, but smooth, then the right side of the graph would twist down sharply. It appears that it could be approximated by $(30-E)2^{E-30}$. Not shown; it would be nice to show this graph.

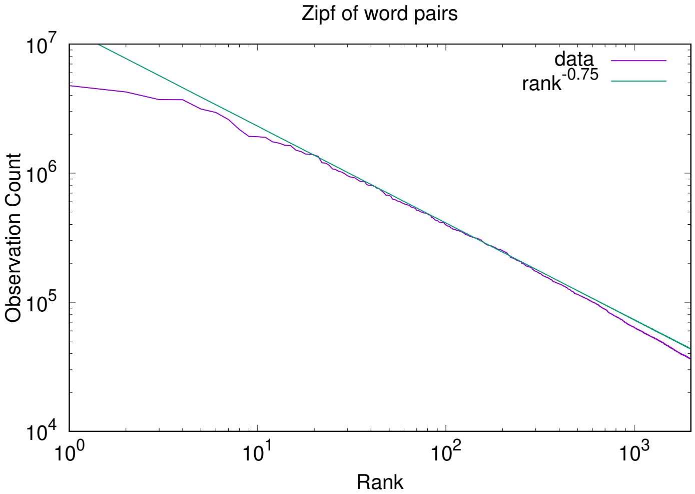
For comparison, below left is the conventional Zipf distribution graph, and, on the right, the same graph flipped along the diagonal, together with density of states from above.





Under-sampling

The humpback shape appears to be due to an under-sampling effect. This is exposed and explained in the next few sections. Due to a finite sample size, it appears that the only pairs that are sufficiently sampled are those up to a rank of about 1200. After that, pairs are under-sampled. The result of that under-sampling is a humpback shape, as seen above; the top of the hump is where the under-sampling begins. This suggests that the eyeballed fit is incorrect, and that the Rank distribution should be considered only up to 1200. This is shown below.



This time, the slope is different: it is 0.75, which is, umm, err, I guess its a "small world" slope. This is no longer the canonical Zipf slope of 1.0. This raises the question: how many of the graphs in the earlier parts of the diary are compromised, as being mixtures of under-sampling and "actual effects"? This also raises the question: aren't all learning effects always driven by an under-sampling? That is, isn't one always doomed to under-sample? How can one know this, and how can one take this into account?

Whence this magic number 1200?

Top-ten Word Pairs

Given the above discussion about under-sampling, it is hard to avoid noticing that the top-ten word-pairs appear to follow an eve flatter slope. What does this mean? Clearly, they are not under-sampled, and so the flatter slope needs to have some more sophisticated explanation.

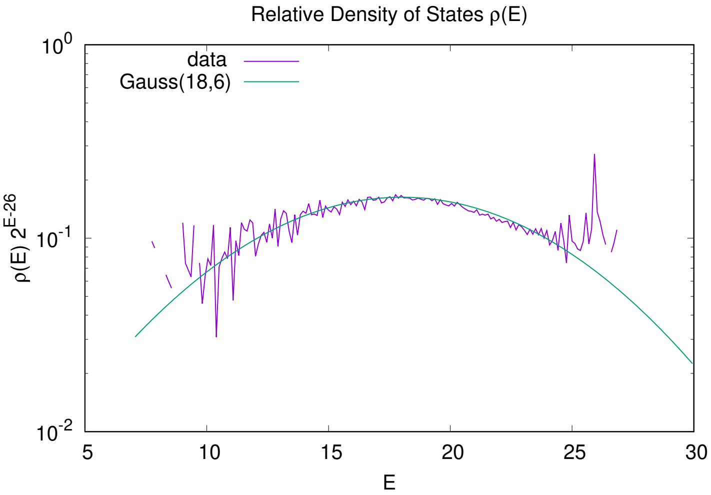
The top-ten most frequently observed word-pairs are shown in the table below:

Rank	Count	Pair (word <<> word)
1	4765096	, <<>> and
2	4254477	###LEFT-WALL### <<>> ,
3	3714944	###LEFT-WALL### <<>> .
4	3705739	of <<>> the
5	3141824	- <<>> -
6	2951005	###LEFT-WALL### <<>> the
7	2603823	, <<>> the
8	2177390	the <<>> of
9	1926906	in <<>> the
10	1915335	, <<>> ,

The “more sophisticated” explanation might be this: this is a finite-sentence-length effect. That is, the nature of human understanding is that we have a limited attention span, and a limited short-term memory. Sentence lengths, and the use of punctuation must accommodate these limits. Thus, sentence starters and sentence enders, and commas, for phrase identification, should appear at a constant rate, rather than at a Zipfian rate. And that is indeed what the above seems to confirm. OK, so that’s an interesting discovery, its new to me.

Relative Density of States

OK, so, due to under-sampling effects, there is a hump. Let’s look at the hump more closely.[4]



The above shows $\rho(E) \times 2^{E-26}$ and an eyeballed Gaussian. The relative factor of 2^{E-26} removes the dominant slope. It’s $E-26$ instead of $E-30$ so that we can draw the Gaussian without any normalization. That is, the Gaussian is just

$$G(\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma}} \exp - \frac{(E-\mu)^2}{2\sigma^2}$$

without any further normalization. It’s drawn for $\mu = 18$ and $\sigma = 6$. There is no theoretical reason (that I am aware of) to expect a Gaussian here.

Properly, we should be fitting to a log-normal distribution, since E is necessarily positive-definite. However, this far from the origin, the log-normal and the normal distributions look almost identical. Whatever, its an OK fit to a bulging, noisy graph.

What does it mean? I dunno. It means that we need a theoretical framework for handling the under-counting phenomenon above.

[4] See ‘p6-density/density.gplot‘.

This Gaussian is probably (almost surely) related to the Gaussian appearing in the MI scores, below. But I do not know how to derive one from the other. (Too lazy to figure it out.)

Physical Interpretation

Clearly, the density graph suggests that the total energy

$$\int E\rho\left(E\right)dE$$

is unbounded. Of course, it is finite for this particular dataset, but the trend suggests that if a trillion word-pairs were observed, then the high-end of the graph would be at $E=40$ instead of at $E=30$. Thus, this is not a "physical" system of finite energy, in the conventional sense.

The root cause of this is that the vocabulary is unbounded. As more and more text is observed, more and more vocabulary words are encountered, and there appears to be no upper limit (geographical place-names, foreign loan-words, given names, imaginative sales terms, children's nonsense words, portmanteaus, ...) As a result, the number of distinct word pairs also grows, in an unbounded fashion, as the number of observations increase.

Thus we take the size of the vocabulary to be countable infinity and denote it as $\mathbb{N}$ the natural numbers. The space of all strings (sentences) of length n is then the Cartesian product

$$\mathbb{N}\times\cdots\times\mathbb{N}=\mathbb{N}^{n}$$

It would be nice to be able to assign a measure μ to this space, but even this is problematic. Consider the cylinder set $(*,\cdots,*,w_{k},*,\cdots,*)$ of a word w_{k} at location j in the middle of all sentences of length n. Denote the probability as $\mu_{j}\left(w_{k}\right)$. For English, and for many languages, the probability of observing a word is mostly independent of it's location in the sentence, so drop the subscript j and just write $\mu\left(w_{k}\right)$ as the probability of observing a word (or just define $\mu\left(w_{k}\right)$ as the average over j.) This is the measure of the cylinder set $(*,\cdots,*,w_{k},*,\cdots,*)$.

For this to be a proper probability, we expect that we should be able to write

$$1=\sum_{k}\mu\left(w_{k}\right)$$

which is an eminently desirable property of any measure. But we are not so lucky: the distribution of words is Zipfian, with exponent 1, and so this sum is logarithmically divergent. That is, the Zipfian distribution of individual words tells us that

$$\mu\left(w_{k}\right)\approx\frac{1}{k^{s}}$$

for exponent s, and experimentally, it is well-known that for natural language, $s\approx1$ and is typically measured to be 1.01 to 1.05 for datasets with millions or billions of words. For the infinite vocabulary $\mathbb{N}$, we are forced to take the value of $s=1$ and thus end up with a topological space without a conventional measure on it.

The only good news here is that it is only weakly divergent.

Except the above conceptualization is wrong. Based on the revised results, taking into account the limited sample size (as discussed above, and further below) we have to conclude that in the limit of large sample size, that $s \approx 0.75$. In addition to this, sentences are not unbounded in length (German philosophers proving the rule) and so the actual normalization requirement is

$$1 = \sum_{k<N} \mu\left(w_k\right)$$

where N is an upper bound on "realistic" sentence lengths.

What all this points at is the lack of a theory that takes into account limited sample sizes, as well as taking into account human cognitive effects such as finite sentence lengths, driven by attention span and the limits of short-term memory. Developing such a theory appears to require considerable effort.

The above density-of-states results for word-pairs indicates that the same applies for $\mu\left(w_i, w_j\right)$. This is the same as $p\left(w_i, w_j\right)$, we're just bouncing around in notation, so that μ is the formal measure on the Cartesian product space, for given cylinder sets, while p is the experimentally observed frequentist probability (the Bayesian probability with the trivial prior.)

Dataset Notes

This part is too big to fit in a footnote, so I put it here. At first, attempted to work with the Run-1 dataset 'run-1-marg-tranche-1234.rdb' which contains the marginals. This dataset is painfully large, taking too long to load. It was using 50 GB and swapping like mad after loading 29M of the total 38M pairs. Ouch. Try again with 'run-1-en_pairs-tranche-123.rdb' which should not have any marginals, just the raw counts ... Hopefully, skipping the marginals takes less time to load (?). The dataset stats appear in Diary Part Two, at the very end.

Nope. We really want to have marginals, for assorted reasons. Try again with 'run-1-marg-tranche-123.rdb'.

Config files are in 'Experiment-13'.

Dataset summary:

Property	Value
Filename	run-1-marg-tranche-123.rdb
Dimensions	304085 × 306920
$\log_2$ Dimensions	18.214×18.228
Num Pairs	28184319
$\log_2$ Num Pairs	24.7484
Total Count	985483375
$\log_2$ Total Count	29.8763
RAM Usage to Load	49.7 GB
RAM Usage to Run	62.6 GB
Entropy Total	18.378

The above dataset summary agrees with what is reported at the very end of Diary Part Two (*i.e.* during load, the same numbers are reported.) The "Entropy Total" is the same as defined in earlier diaries:

$$\text{Entropy Total} = -\sum_{i,j} p(w_i, w_j) \log_2 p(w_i, w_j)$$

Recall, as always, that "Num Pairs" is the number of distinct, unique pairs that were observed, while "Total Count" is the how many times those pairs were observed.

Other Densities

Well, OK, so we've done Zipf graphs of all kinds before, but somewhat haphazardly. It's worth redoing these as densities: i.e. bin-counting, with the horizontal axis being $E = -\log_2 p(w_i, w_j)$ and the vertical axis being other assorted quantities. All of these graphs will suffer from the under-sampling issues described above. We don't yet have a good theoretical foundation to deal with the under-sampling. So damn the torpedoes, full speed ahead.

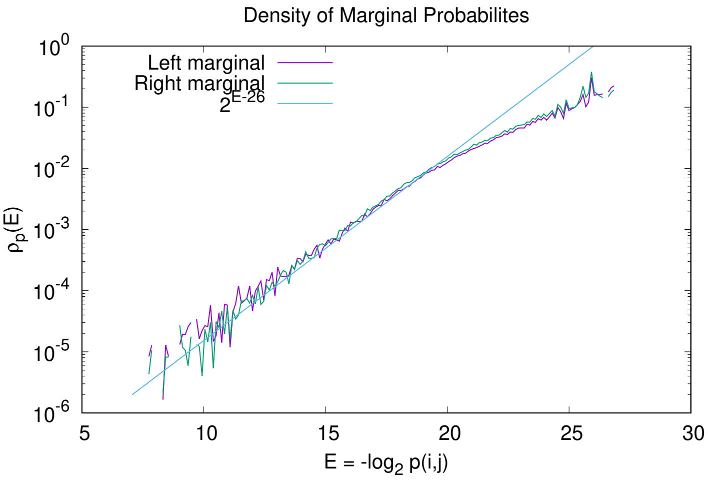
Suitable quantities to plot:

- The pair MI.
- The left and right marginal probabilities $p(w_i, *)$ and $p(*, w_j)$.
- The log left marginal probability $-\log_2 p(w_i, *)$.
- The left fractional marginal entropy $-\frac{1}{p(w,*)} \sum_v p(w,v) \log_2 p(w,v)$.
- The left fractional marginal MI.

These all seem to be pretty, um, boring. I've graphed them all as absolute densities. Perhaps they should be graphs as relative densities. Yet doing so does not seem all that promising; they'll be horizontal lines, right?

Left and Right Marginal Probabilities

First up: the left and right marginal probabilities $p(w_i, *)$ and $p(*, w_j)$.



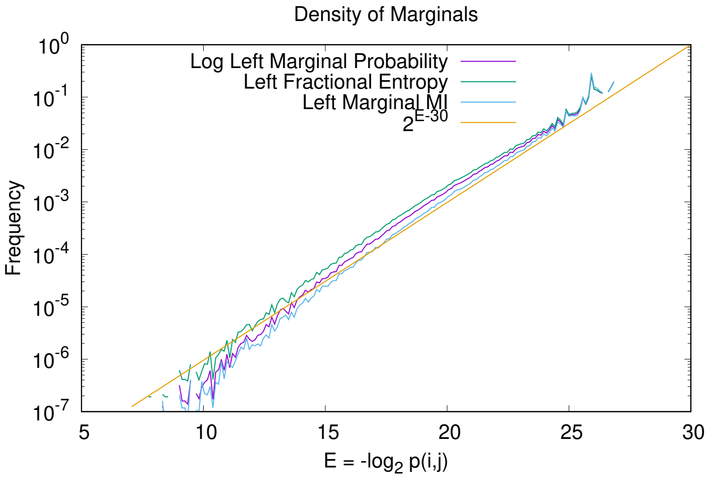
Slope is same as before.

Log Marginal Probability, Entropy and MI

A three-in-one chart:

- The left log marginal probability $\log_2 p\,(w_i, *)$.
- The left fractional marginal entropy $-\left[p\,(w_i, *)\right]^{-1} \sum_v p\,(w_i, v) \log_2 p\,(w_i, v)$.
- The left marginal MI.

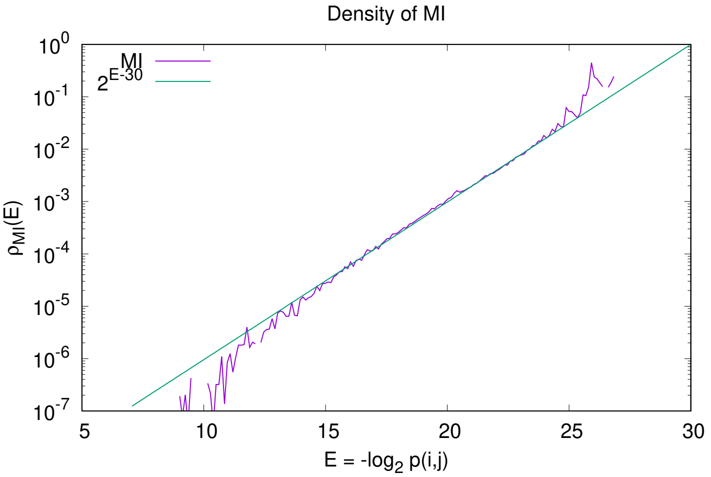
Skip exploring the right, based on the above.



Slopes are muddled. Neither here nor there. Of course, these have to differ slightly in the slopes.

Pair MI Density

The pair MI density $p(w_i, w_j)$. This is *NOT* a marginal!



Slope seems to be nailed exactly. Remarkable. None of the fiddle-faddle of before. Recall as always that the 30 comes from $30 \approx \log_2$ Total Count for this dataset.

Word-Pair Vertex Degree Distributions

Vertex-degree graphs are commonplace in network analysis. Oddly enough, I never really characterized the word-pair sets using more conventional graph-theoretic concepts. Time to make amends.

The collection of word-pairs can be taken to define a set of edges, thus defining a graph. This is a directed graph, but I think this doesn't matter, so will mostly pretend it is undirected. The collection of vertexes will be taken as the left-element of each pair. The edges will be the pairs going from left to right.

The (out-)degree of a vertex is just the number of edges leaving it. We work with out-degrees exclusively, except for a few spot-checks. Basically, the word-pair graph should be approximately symmetric, so there should not be much of a difference in distributions.

The results of this section:

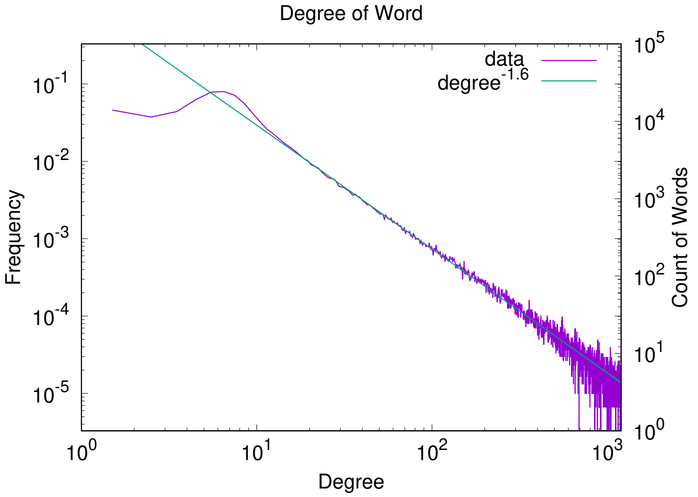
- The scaling of frequency vs. degree goes with a power law of $\gamma \approx 1.6$. This is a "small world" scaling exponent. Under-sampled, infrequent word-pairs belong to a small world!
- There is an interesting sample-size effect, which prevents naive scaling of histogram bin-widths!
- There is no theory to guide one through the sample-size effect, and it is clearly pervasive, affecting pretty much every graph ever drawn, ever, in this diary. It's a foundational effect, that cannot be escaped. Its inherent in this kind of data.
- One can graph all sorts of quantities as functions of the vertex degree. Does not seem to reveal anything noteworthy.

To recap: A vertex is a word, and it's degree is the number of (distinct, unique) word-pairs it occurs in. For the range of $10 \lesssim D \lesssim 1200$, the probability $p(D)$ of observing a word-vertex with degree D goes as $p(D) \sim D^{-1.6}$. This is a small-world scaling exponent; it is far away from being a scale-free network exponent.

Note that this is a direct measurement of the under-sampled parts of the graph. That is, a word that has a small degree is necessarily a word that is observed infrequently. It cannot be a word like "the", which will have a degree of approx 100K (for this dataset). It cannot be a preposition, as these will also have degrees of 50K or more. Even common verbs and nouns are expected to have a gigantic degree.

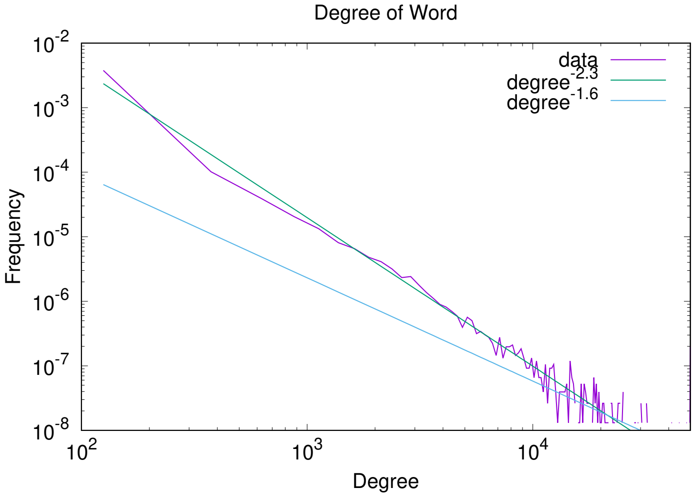
Vertex Degree

The first conventional question is "what is the vertex degree distribution?" This is shown below, a graph of the normalized frequency of a word, and it's out-degree. The degree ranges as high as 300K with significant counts up to 50K. The graph only shows degree up to 1200. There are 1200 bins in this graph, so each different degree gets it's own bin.



The eyeballed fit has frequency $= (1.1/\text{degree})^{1.6}$ and so that exponent is well below typical scale-free networks. The raw counts are shown on the right y-axis, *i.e.* un-normalized. The point of drawing it this way is that we see on the right where the count drops to one (and to zero).

Something unexpected happens if we go deeper. There are gaps between words with high degrees, and it seems like it should be reasonable to bin them together. The graph below shows degree out to 50K, collected into 200 bins. Thus, each bin is 250 degrees wide. The slope is remarkably different:



I think this is purely an sampling effect. In principle, the slope should not have changed. Here's a quick sketch. If the original distribution is

$$f_n = \left(\frac{a}{n}\right)^{\gamma}$$

then a re-binning into constant-sized bins of size k is given by

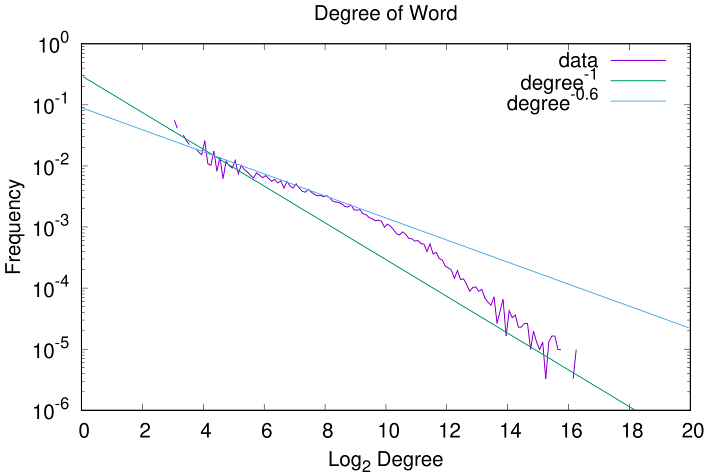
$$\begin{aligned}
g_m =& \sum_{m(k-1)<n\leq mk} f_n \\
=& \left(\frac{a}{k(m-1)+1}\right)^{\gamma} + \cdots + \left(\frac{a}{k(m-1)+k}\right)^{\gamma} \\
=& \left(\frac{a}{k(m-1)}\right)^{\gamma} \left[\frac{1}{\left(1+\frac{1}{k(m-1)}\right)^{\gamma}} + \frac{1}{\left(1+\frac{2}{k(m-1)}\right)^{\gamma}} + \cdots + \frac{1}{\left(1+\frac{k}{k(m-1)}\right)^{\gamma}}\right] \\
\approx& \left(\frac{a}{k(m-1)}\right)^{\gamma} \left[k - \frac{\gamma}{k(m-1)}\left[1+2+\cdots+k\right]\right] \\
\approx& \left(\frac{a}{k(m-1)}\right)^{\gamma} k \left[1 - \frac{\gamma}{2(m-1)}\right] \\
\approx& Cm^{-\gamma}
\end{aligned}$$

where the approximations $k \gg 1, m \gg 1$ are made. That is, there is a change in the overall normalization, and the early part of the slope, for small m is reduced, but the overall exponent is not affected. Yet this is given lie to by the figure above. So what goes wrong? It is an sampling effect. For large n, most of the f_n are not as given above, but are zero. The fraction of the time that they are zero is determined by the sample size, and they are zero often enough that the overall slope is changed. The net effect of sample size could be computed. Just right now, it does not seem to be a worthwhile exercise. XXX TODO. Do this anyway. This should be done. This seems like a foundational part of the overall theory.

Let's repeat the calculation above with an explicit log scale. This gives

$$\begin{aligned}
g_m = & \sum_{m \leq \log_2 n < m+1} f_n \\
= & \sum_{2^m \leq n < 2^m} \left(\frac{a}{n}\right)^\gamma \\
= & a^\gamma \sum_{1 \leq j < 2^m} \frac{1}{(2^m + j)^\gamma} \\
\approx & \frac{a^\gamma}{2^{m\gamma}} \cdot 2^m \left(1 - \frac{\gamma}{2}\right) \\
= & C 2^{m(1-\gamma)}
\end{aligned}$$

Thus, since we saw $\gamma \approx 1.6$ in the earliest figure, we expect a slope of $\gamma = 1 \approx 0.6$ in the equivalent log figure. This is shown below.



That initial slope is valid up to $n \approx 1200$ or $\log - \frac{1}{p(w,*)} \sum_v p(w,v) \log_2 p(w,v)_2 n \approx 10$, after which a sharper slope sets in due to the sample-size effect. This gives the figure an overall hump-back shape.

High Degree Vertexes

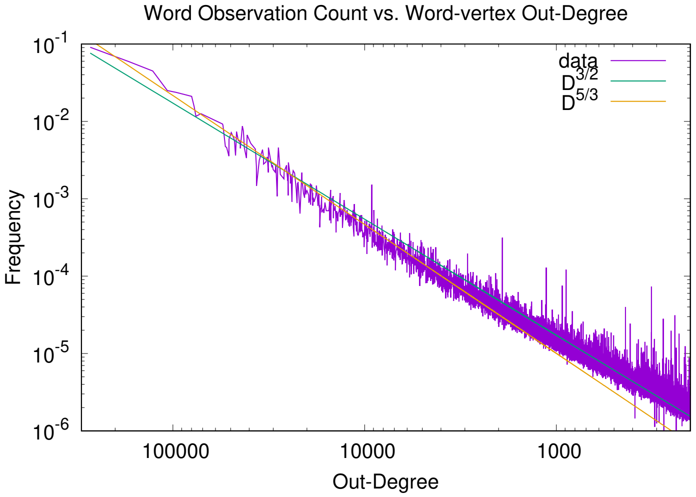
Where is this effect coming from? The table below shows the 60 words with the highest out-degree. This offers a glimpse of what is going on at the other end of the vertex-degree scale.

1-20		21-40		41-60	
Word	Degree	Word	Degree	Word	Degree
LEFT-WALL	269970	is	45618	her	31763
,	169643	for	44599	are	30478
the	127386	on	44496	all	30246
of	107029	I	43449	one	30035
and	96207	"	43411	their	29283
to	79874	he	42884	they	29031
a	77490	at	42390	(	28581
-	75847	from	41338	him	28338
in	71456	it	41000	you	28166
was	55051	had	39295	been	27532
that	54780	be	37145	who	26380
;	53249	or	36689	so	26314
The	52418	:	36647	He	25543
_	51059	which	35451	my	25535
with	50988	not	34088	when	25177
.	49642	but	33683	”	25160
“	48150	were	33675	—	25054
as	47770	this	32726	she	24973
by	47135	have	32202	up	24385
his	46552	an	31814	into	24374

Clearly, there are large gaps in degree between each of these. Clearly, as the size of the corpus is increased, the degree of each of these will increase, and the size of the gaps between them will also increase. The overall order and distribution should not substantially change.

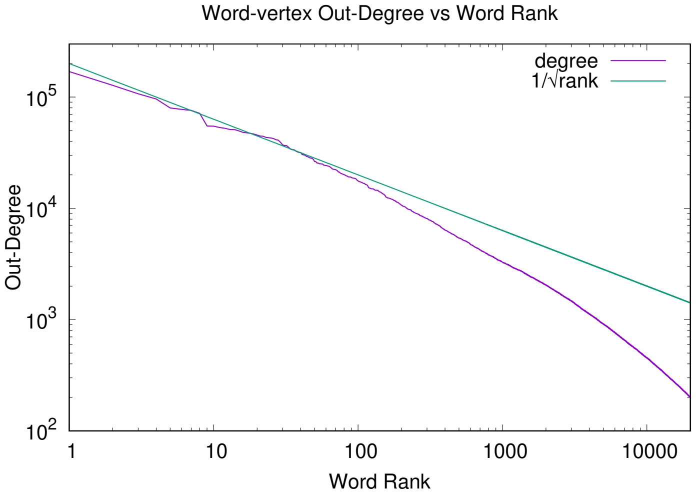
In short: attempting to bin-count the above leads to misleading confusion. Naively, bin-counting is about smoothing variation. But, as this table makes clear, such “smoothing” is actually averaging in empties. It’s not “smoothing”, its altering the slope.

Here is the table above, directly visualized:



This shows the frequency with which a word was observed, vs. it's out-degree, which is exactly what the table is depicting. Note the "reversed" x-axis. Two eyeballed fits are presented: $D^{3/2}$ and $D^{5/3}$ for the out-degree D. The frequency is, as always, the total number of observations of that particular word, normalized by the total number of observed words. The integral under the curve is one.

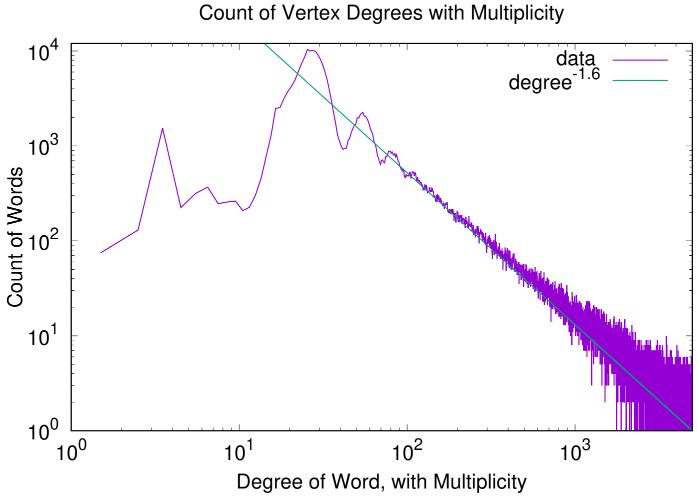
For completeness, here's a traditional degree vs. rank graph:



Notice that initially, the vertex degree falls off as $1/\sqrt{\text{rank}}$, which seems to be a very traditional slope for this kind of graph of network degrees. I still don't understand why, despite a half-a-decade of seeing this one-over-sqrt distribution. It's everywhere: see the wikipedia page rank, see the agi-bio genome and reactome distributions I've graphed elsewhere.

Weighted Vertex Degree

Same as above, but showing the weighted vertex degree, i.e. “with multiplicity”. That is, if each edge was observed N times, then it is treated as if there were N distinct edges.



There is a distinct oscillatory behavior at the center-left. It is perhaps some strange artifact, having to do with the fact that 24 parse-trees are sampled, given that the first, most prominent peak occurs at 24, and later peaks are perhaps at multiples of 24.

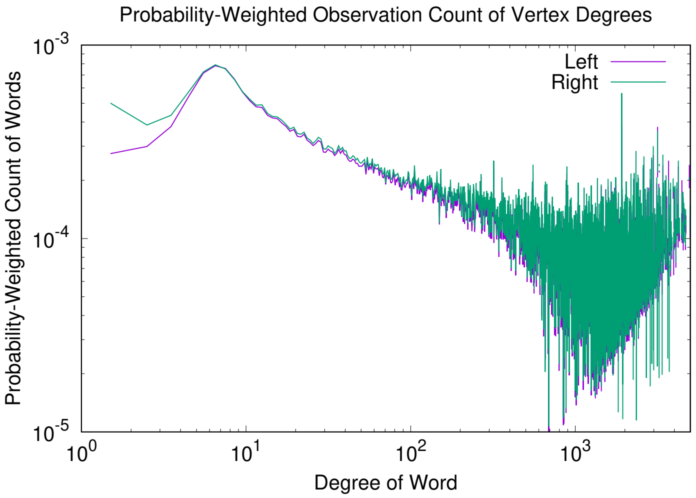
Other weighting schemes

The following sequence of graphs use weighting schemes that are perhaps difficult to understand, but seem worth exploring. Their physical interpretation is challenging and the significance is unclear. Do it anyway, just to say we covered all the bases.

In all of these, the horizontal axis shows the edge degree of the word, without multiplicity, so, the number of unique word-pairs that a word participates in. The y-axis, however, uses weighted counting, with different weights. That is, if a word has an edge degree of 42, then instead of counting it exactly once, it is counted with a weight (mass) $m \neq 1$. Graphs are shown for

- Mass $m = p(w,*)$ the right marginal probability for word w.
- Mass $m = p(*,w)$ the left marginal probability for word w.
- Mass $m = \log_2 p(w,*)$ the right marginal log-probability for word w.
- Mass $m = \sum_v p(w,v) \log_2 p(w,v)$ the right marginal entropy for word w.
- Mass $m = MI(w,*)$ the right marginal MI for word w.

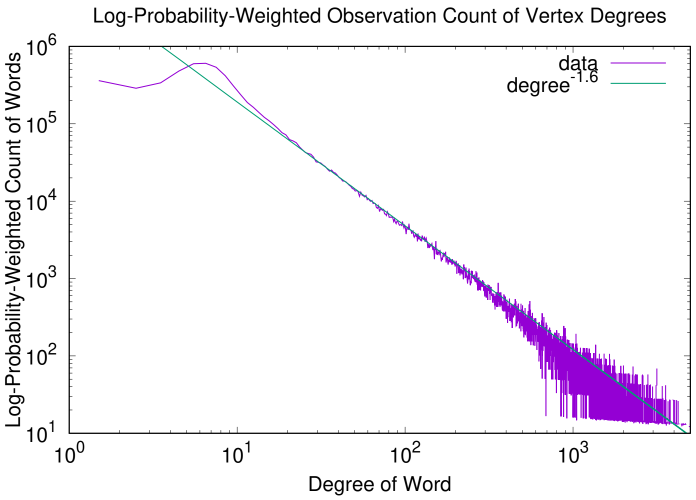
Weighted by Marginal Probability



The vertical axis totals the marginal probability for that word. That is, instead of adding 1 for each observed edge (the support), it adds $p(w,*)$, the left-marginal probability, or $p(*,w)$, the right marginal probability. So, for example, consider a vertex of degree 5. There might be 10K such vertexes in this dataset. However, each such vertex might be observed 30 or 50 times, and so (for left-marginal counts) we would see 300K or 500K on the y-axis here.

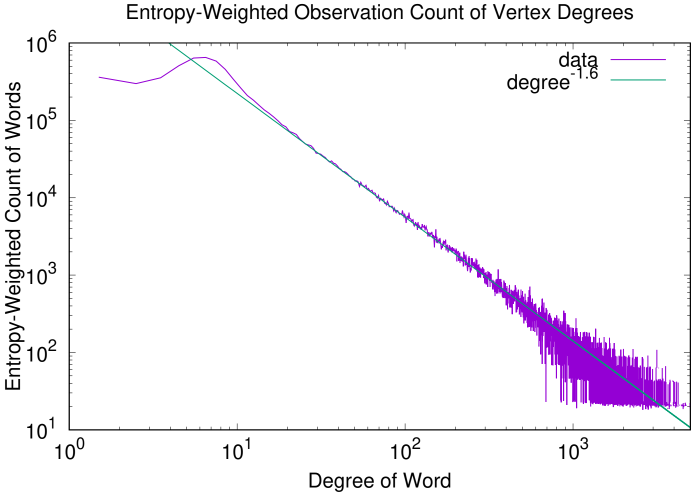
Weighted by Marginal Log Probability

Weighting by the marginal log probability straightens things out:



The vertical axis totals the log marginal probability for that word. The weight is $-\log_2 p(w,*)$.

Weighted by Marginal Entropy

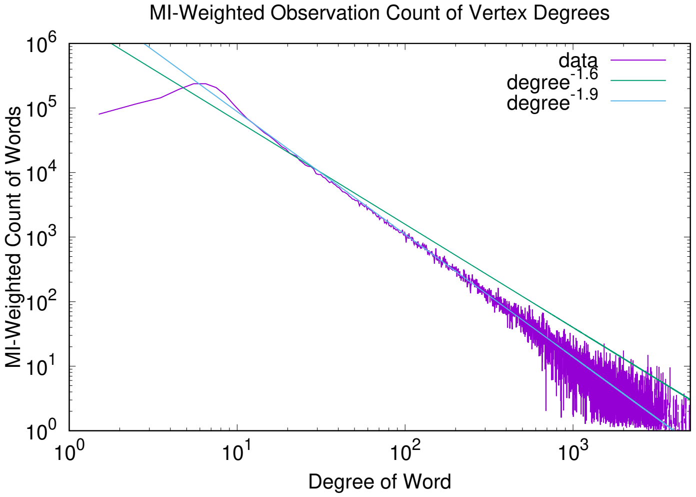


The vertical axis totals the marginal fractional entropy for that word. The weight is

$$-\frac{1}{p(w,*)}\sum_v p(w,v)\log_2 p(w,v)$$

Clearly, this graph is nearly identical to the above; it is shifted ever so slightly upwards.

Weighted by Marginal MI



The vertical axis totals the marginal fractional MI for that word. The weight is

$$\frac{1}{p(w,*)}\sum_v p(w,v)\log_2 \frac{p(w,v)}{p(w,*)\,p(*,v)}$$

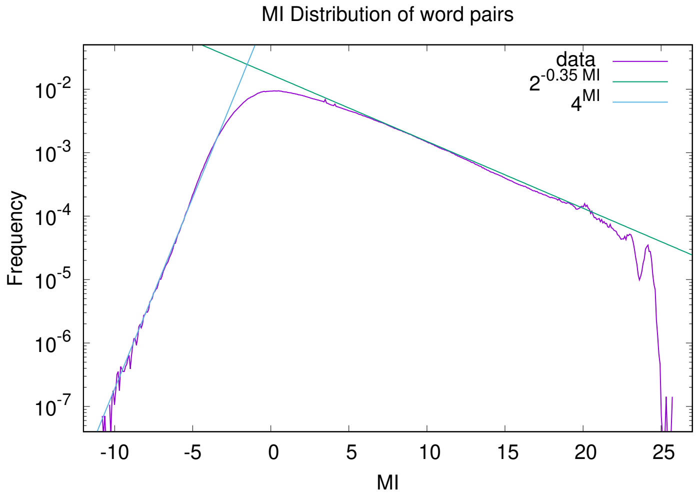
This graph has the same general shape as the earlier ones, but has a distinctly different slope: its 1.9 instead of 1.6.

Word-Pair MI Distribution

We've graphed the MI distribution many times before. Notably, the "Word-Pair Distributions" document details these. But since we're on a roll here, lets redo it with the same dataset as all the other graphs.

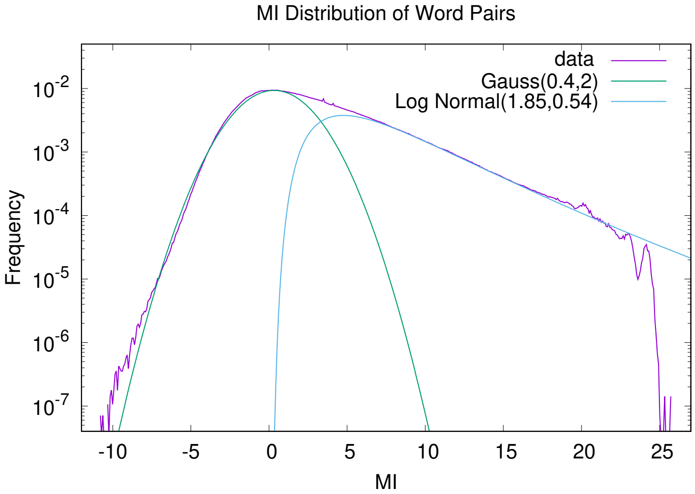
The big new news for this section is that I think I finally understand the nature of this distribution. It is composed of two parts. One part is a Gaussian, centered more or less on an MI of zero. This Gaussian is purely due to selections of word-pairs that contain no syntactic information. Subtracting this leaves behind the word-pairs with the actual syntactic information. That distribution seems to be log-normal, i.e has strictly-positive MI.

The distribution of word-pair MI is shown below. As before, this is for dataset 3, which contains 28 million pairs. The MI is sorted into 500 histogram bins.



The distribution is clearly not symmetric. The two sides appear to be bounded by straight lines, with slopes as in the legend. Pairs with the highest MI are observed very infrequently.

Here's the same data, but with a different fit:



Why this shape? Here's a guess. If word-pairs are chosen completely at random, and the number of sampled pairs is much smaller than the total possible pairs, then one obtains a Gaussian distribution. Such a distribution is centered on a small but positive MI, due to sample-size effects. For larger samples, the mean tends to zero. Thus, perhaps the left-hand-side of this figure is just a Gaussian.

Now, we are not selecting word-pairs "at random", but we are sampling all possible word-pairs over a short region of text. These are sampled uniformly, by uniform selection of random MST trees. Many of the sampled word pairs will not be linguistically-related, but instead just accidentally near each other; they are near each-other for semantic reasons, not syntactic reasons.

Taking this Gaussian to be "common-mode noise", and subtracting it, leaves an excess of word pairs with positive MI, having a peak near $MI \sim 4$. The straight-line slope on the right suggests that the excess can be described by a log-normal distribution. Again, an eye-balled, imprecise fit is shown. These two, summed together, model the observed distribution almost perfectly. Perhaps a formal expression for the common-mode noise is easily derived, given a fixed vocabulary size and number of samples. An attempt to get this is made further below.

Theories of why the remainder would be a log-normal distribution are unknown to the author.

Pairs with the highest MI are observed very infrequently. The highest observable MI value is directly related to the sample size: it is a bit below the log of the number of observations. Thus, the sharp drop on the right side is purely a sample-size effect. Trimming does not appreciably change the shape of this distribution, other than to eliminate the very highest MI values.

This distribution is not language-specific; a nearly identical distribution is seen for Chinese Mandarin Hanzi pairs.[5]

FWIW, note that the left-hand-side slope might be more linear than it is parabolic.

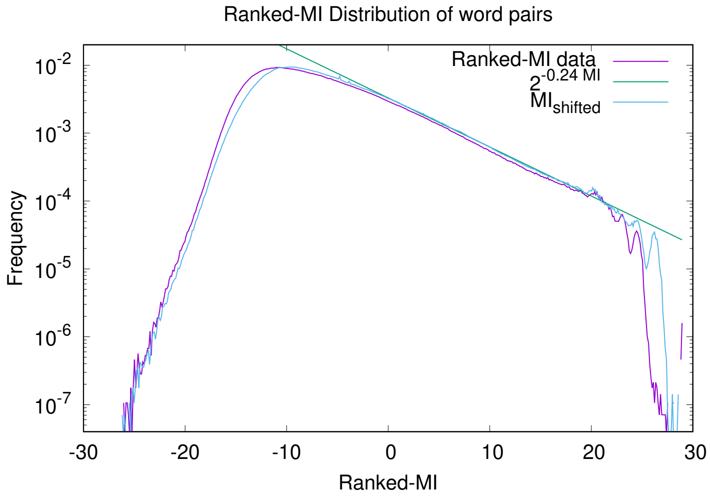
[5] See "Word-Pair Distributions", page 18.

In this case, the left-hand side might also be modelled with a log-normal distribution, this time, mirrored. In this case, the whole idea becomes a bit more of a mirage: we have two straight lines, and the interpolation between them is accomplished by combining a pair of offset and mirrored log-normals. This does not change the overall conclusion: the bulk of the low-MI pairs are due to random noise from sampling, while the high-MI pairs encode syntactic information.

In practice, when given any particular word pair, how can we know which group it belongs to? Is it syntactic, or not? The old rule of thumb was that pairs with MI=4 or greater were meaningful, and those below were junk. The graphs above validate this rule of thumb. Forcing a hard-cut at MI=4 removes most of the zero-mode Gaussian. But we've known for eons that MI=4 is somehow magical; now we finally have insight as to why.

Ranked-MI Distribution

Just for giggles, here's the ranked-MI distribution:



Recall the ranked-MI is defined as

$$MI_{\text{ranked}}(w,v) = \log_2 \frac{p(w,v)}{\sqrt{p(w,*)\,p(*,v)}}$$

Superimposed on this graph is the distribution of the regular MI, multiplied by 1.5 to get the width correct, and shifted by 10 to the left, to get the zero correct. Apparently, ranked-MI does not alter the distribution. I wonder what it does, if it were used for MST/MPG parsing ...

Oh, hang on. the Ranked-MI is just 1/2 of the "variation of information", see Diary Part Two, page 54. Oh huh. OK, so I have to go back into the diary, and amend all of the entries to reflect this conventional name. Yow!

Neutral MI Distribution

Attempted theoretical calculation of the MI distribution that would result if word pairs were chosen at completely at random. There are two distributions of interest. One uses a uniform distribution of the vocabulary words, the other a Zipfian distribution.

XXX FIXME Everything below is incomplete, incorrect, wrong. I'm too lazy to figure out why.

Uniform distribution

Assume a vocabulary size of N. A random word-pair consists of two random, uniformly-weighted draws from this vocabulary. We take the order of the draws as being important; thus, any given word-pair has a chance of $1/N^2$ of being drawn. Consider M pair draws, with $N \ll M \ll N^2$.

The chance of a given pair being drawn once is CM/N^2 with C a normalization constant to be determined. Basically, it can be drawn the first time, or the second time, or the third time ... etc. but never twice. The chance of it being drawn twice is $CM\left(M-1\right)/2N^4$ and so now we have the usual combinatorics. The chance of observing a word-pair $\left(a,b\right)$ a total of K times is

$$p\left(K|a,b\right)=\frac{C}{N^{2K}}\binom{M}{K}$$

Right(?)

Zipfian distribution

Repeat the above, with a non-uniform distribution. Each word is distinguished by it's ordinal k so that we have words w_k for $1\le k\le N$. The probability of drawing word w_k is then $p\left(w_k\right)=Ak^{-\gamma}$ for $\gamma\approx 1$ and A a normalization constant, so that $1=\sum_{k=1}^{\infty}p\left(w_k\right)$. The probability of drawing a pair $\left(w_i,w_j\right)$ is then $p\left(w_i,w_j\right)=p\left(w_i\right)p\left(w_j\right)$ since the probabilities are completely independent of one-another.

Now we have to iterate this experiment M times. The probability of drawing a given pair K times is then

$$p\left(K|j,k\right)=C\left(p\left(w_i,w_j\right)\right)^K\binom{M}{K}$$

Write $x=p\left(w_i,w_j\right)$ for short, then the normalization is

$$f\left(w_i,w_j\right)=C\sum_{K=0}^{M}x^K\binom{M}{K}=C\left(1+x\right)^M$$

Right??? I'm confused. Now, since $\varepsilon=Mx\ll 1$ we can write

$$f\left(w_i,w_j\right)=C\left(1+Mp_ip_j+\mathscr{O}\left(\varepsilon^2\right)\right)\approx C\left(1+Mp_ip_j\right)$$

and so the marginal is

$$f\left(w_i,*\right)=\sum_j f\left(w_i,w_j\right)\approx CN+CMp_i$$

but $N \gg Mp_i$ by assumption, so $f(*,*) = 1 \approx CN^2$ and so

$$f(w_i,*) = \frac{1}{N}$$

which seems wrong, so I made a mistake above!?

Word-Disjunct MI Distribution

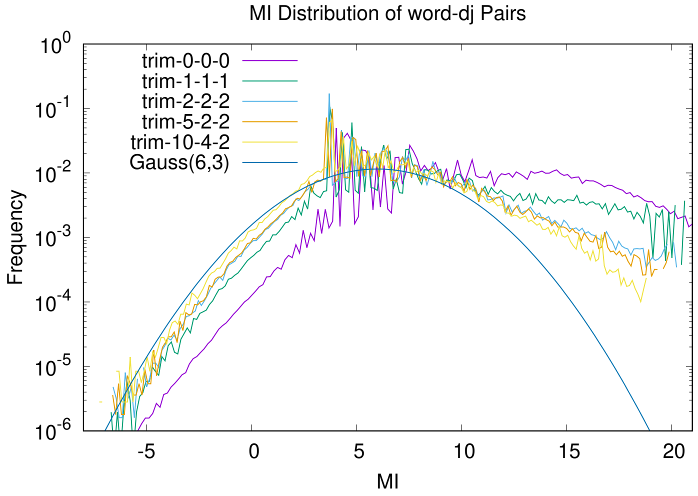
This revisits earlier results on connector-set MI distributions; see page 22 of the “Connector Sets Distributions” document.

This revisit will work with ‘r4-mpg-marg.rdb‘ which appears on page 4 of Diary Part Three. Actually, this is just a copy of ‘run-1-en_mpg-tranche-123.rdb‘ with marginals in it.

Although the older ‘r4-trim-*.rdb‘ were “trimmed”, the self-consistency checks were crappier. So we re-do the final self-consistency checks, using the code in ‘scm/gram-class/cleanup.scm‘ to get fully consistent results. The updated dimensions are in the table below. The files are ‘r4-trim-10-4-2-djmi.rdb‘, etc. but are shortened in the column labels.

	full	1-1-1-djmi	2-2-2-djmi	5-2-2-djmi	10-4-2-djmi
N_L= words	377553	47708	12800	7586	4867
N_R= dj	25698949	1587889	414713	357457	169277
D_{Tot}= size	28436901	2049074	622378	556413	356298
N_{Tot}= obs	36389195	9736866	6496202	6128265	5279297
$\log_2 N_{\text{word}}$	18.5263	15.5419	13.6439	12.8891	12.2488
$\log_2 N_{\text{dj}}$	24.6152	20.5987	18.6618	18.4474	17.3690
$\log_2 D_{\text{Tot}}$	24.7613	20.9665	19.2474	19.0858	18.4427
sparsity	18.3803	15.1741	13.0582	12.2507	11.1751
rarity	3.19050	2.89623	3.09463	3.41753	3.6338
$\log_2 N_{\text{Tot}}/D_{\text{Tot}}$	0.35575	2.24849	3.38373	3.46125	3.8892
Entropy	24.1003	19.4863	17.7107	17.5078	16.8745
Left Entropy	23.4936	18.3455	16.4170	16.1626	15.3793
Right Entropy	10.1570	7.93672	7.28017	7.26809	7.25804
MI(w,dj)	9.5504	6.7959	5.9865	5.9228	5.7628

What do the distributions look like? Here they are, all plotted on a single graph. These are histograms. There are 200 bins grand-total. First graph weights each Section equally.

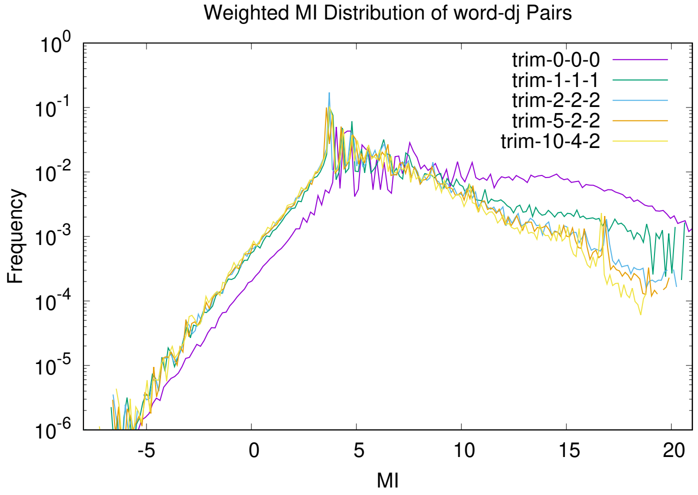


A generic Gaussian is superimposed on the image, centered at about the average MI of each of these. Clearly visible is an excess of counts at high MI values. Clearly, the more one trims, the more of these are removed.

Earlier disjunct graphs, namely in the "Connector Set Distributions" document. page 22, were pure Gaussian, without the excess. Why? Is the excess due to some of the garbage (the failed quote-escapes) in this dataset? Or is it real?

Earlier, for the word-pairs graph, the Gaussian was interpreted as the zero-mode, and was centered at more-or-less zero. But what is this beast? Is it a zero-mode? Is it meaningful data? Or is just the high-MI stuff "meaningful"? How can we know?

Again, below, this time each Section is weighted according to it's frequency.



In what sense is this weighted distribution “more accurate” than the unweighted one?

Mutual Information vs. Laplacian

The discrete Laplacian on a 1-dimensional grid is a tri-diagonal matrix of the form

$$\Delta x_i = 2x_i - x_{i-1} - x_{i+1}$$

The pair-wise mutual information

$$MI(u,v) = \log_2 p(u,v) - \log_2 p(u,*) - \log_2 p(*,v)$$

Inserting a factor of 1/2 gives the ranked-MI

$$MI_{\text{ranked}}(u,v) = \log_2 \frac{p(u,v)}{\sqrt{p(u,*)\,p(*,v)}}$$

The discrete Laplacian somehow feels “similar” to the ranked-MI, but how? Can this be developed?

(Dead-end?) Ideas

Here are some suggestive ideas that don’t sem to quite get traction.

The *MI* “feels like” some kind of re-normalized propagator, where the $\log_2 p(u,*)$ feel like vacuum corrections; but how this could be is opaque.

The point (u,v) feels like a point in a base-space, and the $(u,*)$ and $(*,v)$ feel like two different fibers above the point in the base space. The summation is happening on the fibers. That is, we’ve defined

$$p(u,*) = \sum_w p(u,w)$$

so that, first, we take the fiber sum, then the log, and then compare a point in base-space to it’s two fiber-sums. That is, the ranked-MI *is* a kind of discrete Laplacian, but it’s over a weird fibered space; its a comparison over fibers. The generalization of this would be a funky Hamming-fibered Laplacian, so that, for triples,

$$\nabla(u,v,w) = \log_2 p(u,v,w) - \frac{1}{3}\left[\log_2 p(u,v,*) + \log_2 p(u,*,w) + \log_2 p(*,v,w)\right]$$

and so on. (For one-dimensional fibers). So, conceptually, MI and ranked-MI are a kind of difference equation; they are kind of like fibered Laplacians, but ... what can we do with this insight? What can be constructed?

Hamming Laplacian

Consider the very high-dimensional difference equation

$$-\Delta E(u,v) = \log_2 p(u,v) - \frac{1}{2(N-1)}\sum_{w\neq v}\log_2 p(u,w) - \frac{1}{2(N-1)}\sum_{w\neq u}\log_2 p(w,v)$$

where N is the size of the vocabulary, viz. $N = \sum_w 1$. Using terminology from Chapter 6, this was being called "the energy", via analogy to the Boltzmann distribution. That is,

$$E(u,v) = -\log_2 p(u,v)$$

and so the difference eqn is

$$\Delta E(u,v) = E(u,v) - \frac{1}{2(N-1)}\left[\sum_{w\neq v} E(u,w) + \sum_{w\neq u} E(w,v)\right]$$

This difference equation is rightfully called a discrete Laplacian on a high-dimensional space. That this is the correct name can be seen as follows.

Basically, we're fixing a point (u,v) in the high-dimensional space $N\times N$ and then we are differencing to all of it's nearest neighbors. This is confusing because we really should have started with single words. Consider the observation frequency of a single word $p(w)$ and define $E(w) = -\log_2 p(w)$. Experimentally, we don't track these values (why not? they've never seemed useful. But perhaps we should revisit.) A single word w can be thought of as a coordinate or direction in a high-dimensional space, so that that $(w)\in N$ is a location, a single point in that space. All the other words provide the other coordinates, so that the (u)'s are all of the nearest-neighbors of (w).

In this case, the Laplacian really is clear and unambiguous: it is

$$\Delta E(w) = E(w) - \frac{1}{N-1}\sum_{u\neq w} E(u)$$

This is the conventional N-dimensional finite-difference Laplacian[6], where we've taken the liberty of dividing by N because it is so large. If this still feels odd: bear in mind that all points (u) are nearest neighbors of the point (w). The space itself is a simplex: all N points are equidistant from all the others. This is just Hamming distance one for a string of symbols that is one symbol long.

For word-pairs, fixing a word-pair (u,v) and then asking what all the Hamming distance-one pairs are, these are precisely $\{(u,w) : w\neq v\}\cup\{(w,v) : w\neq u\}$. That is the set involved in the definition of the pair-Laplacian. Should we call this the Hamming-Laplacian? The generalization to N-grams is obvious; we don't need this generalization (yet; we'll need something like of for the disjuncts! As disjuncts are just skip-grams in disguise.)

Still, worth formalizing it. Given a k-gram $\sigma = (w_1,\cdots,w_k)$, define the set of all k-grams that are Hamming-distance zero or one from σ as

$$\{s\} = (*,w_2,\cdots,w_k)\cup(w_1,*,w_3,\cdots,w_k)\cup(w_1,\cdots,*)$$

[6] See Wikipedia, Discrete Laplace Operator

The Hamming-Laplacian is then

$$\Delta E(\sigma) = E(\sigma) - \frac{1}{k(N-1)} \sum_{\tau\in\{s\};\tau\neq\sigma} E(\tau)$$

The denominator is simply the number of terms in the summation.

Note that nothing specifically calls out for E in the above definition: the Hamming-Laplacian can be applied to any function $f(\sigma)$. Also, note that this extends to the entire sigma-algebra. The Hamming distance provides a graph structure to the sigma algebra, indicating which elements of the algebra are nearest-neighbors. I am not aware of any theoretical results on Hamming Laplacians on sigma algebras. Save one, an obvious one: $\Delta\mu = 0$ where μ is a measure on the sigma-algebra. Now that I see this, this is definitely a very interesting and curious thing to explore!

Caution about notation. Note that

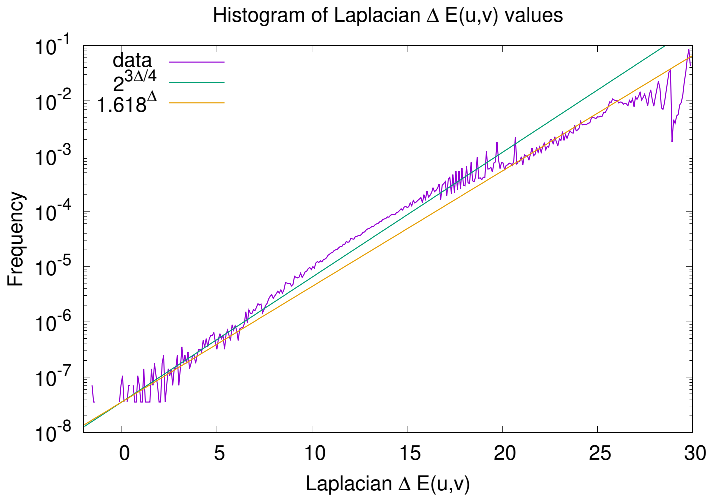
$$E(u,*) = \sum_w E(u,w) = -\sum_w \log_2 p(u,w) \neq -\log_2 \sum_w p(u,w) = -\log_2 p(u,*)$$

Obviously, the log and the sum cannot be interchanged, but using the star notation for wild-card sums makes it tempting to do so.

Experimental exploration

We've not looked at this beastie before. Let's take a look now.[7]

The dataset being used is 'run-1-marg-tranche-123.rdb' – this is well-described elsewhere. It's untrimmed, its got a vocabulary of N=391548 words and a total of 28184319 unique word pairs. These were observed a total of 985483375 times. The graph below shows a histogram of the distribution of $\Delta E(u,v)$ of word-pairs (u,v). There are 400 buckets in the histogram.



[7] The graphs are constructed from the datasets located in the directory 'p6-lapalce'.

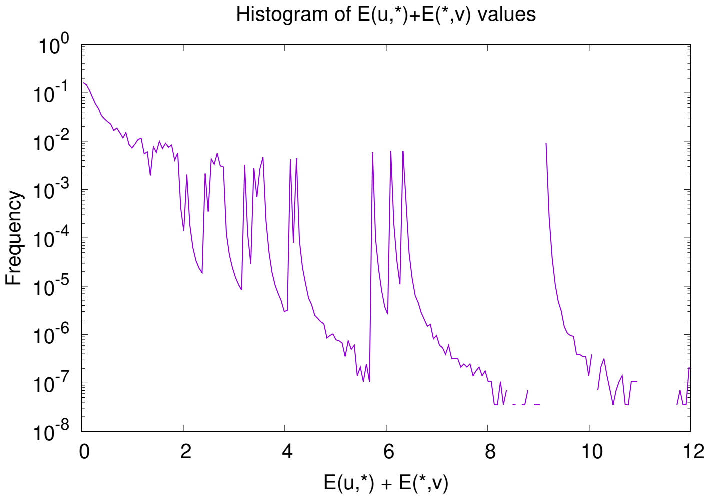
The fit curve is exactly given by $2^{3\Delta E/4}/N_{\text{pairs}}$. That is, the slope is 0.75. Note that $2^{0.75} \approx 1.682$. As usual, the line is bowed, but we don't know why. Just for grins, there is a second fit: some numerology: φ^{Δ} where $\varphi \approx 1.618$ the golden mean. This probably doesn't mean anything, though.

How does the data lead to this graph? Note that $\log_2 N = 18.58$ and that $\log_2 N_{\text{pairs}} = 24.75$ and finally $\log_2 T = 29.88$. The far-right hand side tells us that almost all word-pairs are observed once *i.e.* $E = -\log_2 p = 29.88$ or maybe twice: $E = 29.88 - 1$ or three times: $E = 29.88 - \log_2 3$. At the same time, these words are observed with a far more frequent neighbor: *e.g.* (the, X) for some obscure word X(maybe a typo?), so that although the pair (the, X) is observed only once, the marginal sum $\sum_{w \neq X} E(\text{the}, w)$ is small. Much of this graph is effectively just a reproduction of the earlier $E(u,v)$ graph, including the bow in the middle.

The fiber sums are .. curious. These are (with mild abuse of notation) $E(u,*) + E(*,v)$ or, more precisely,

$$SE(u,v) = -\frac{1}{2(N-1)}\left[\sum_{w \neq v} E(u,w) + \sum_{w \neq u} E(w,v)\right]$$

This is shown in the figure below.



That is, most of the differential corrections are small; the vast majority of them are less than one. So, indeed, we can safely conclude that the distribution of $\Delta E(u,v)$ is indeed very nearly the same as that of $E(u,v)$, as the fiber-sum corrections are small.

This graph looks messy, until one notes that it is approximately self-similar: the right-most limb is repeated on the left, getting progressively smaller; the right-most limb recapitulates the entire graph. I think the cause for this involves words that are see only once, twice, three times. Due to the pair sampling, if a word is seen only once, it will still appear in many pairs: it will be paired with other nearby words.

Conclusion: what have we learned? Nothing really. Interesting ideas, but they don't seem to offer insight that wasn't already there. Nor can I find any useful application for them. What further can be built from this?

The End

This is the end of Part Six of the diary.