CHAPTER

J

Pointwise Mutual Information (PMI)

An alternative weighting function to tf-idf, PPMI (positive pointwise mutual information), is used for term-term-matrices, when the vector dimensions correspond to words rather than documents. PPMI draws on the intuition that the best way to weigh the association between two words is to ask how much **more** the two words co-occur in our corpus than we would have a priori expected them to appear by chance.

pointwise mutual information

Pointwise mutual information (Fano, 1961)[1] is one of the most important concepts in NLP. It is a measure of how often two events x and y occur, compared with what we would expect if they were independent:

$$I(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)} \tag{J.2}$$

The pointwise mutual information between a target word w and a context word c (Church and Hanks 1989, Church and Hanks 1990) is then defined as:

$$\text{PMI}(w,c) = \log_2 \frac{P(w,c)}{P(w)P(c)} \tag{J.3}$$

The numerator tells us how often we observed the two words together (assuming we compute probability by using the MLE). The denominator tells us how often we would **expect** the two words to co-occur assuming they each occurred independently; recall that the probability of two independent events both occurring is just the product of the probabilities of the two events. Thus, the ratio gives us an estimate of how much more the two words co-occur than we expect by chance. PMI is a useful tool whenever we need to find words that are strongly associated.

PMI values range from negative to positive infinity. But negative PMI values (which imply things are co-occurring *less often* than we would expect by chance) tend to be unreliable unless our corpora are enormous. To distinguish whether two words whose individual probability is each 10^{-6} occur together less often than chance, we would need to be certain that the probability of the two occurring together is significantly less than 10^{-12}, and this kind of granularity would require an enormous corpus. Furthermore it's not clear whether it's even possible to evaluate such scores of 'unrelatedness' with human judgments. For this reason it is more common to use Positive PMI (called **PPMI**) which replaces all negative PMI values with zero (Church and Hanks 1989, Dagan et al. 1993, Niwa and Nitta 1994)[2]:

PPMI

$$\text{PPMI}(w,c) = \max(\log_2 \frac{P(w,c)}{P(w)P(c)}, 0) \tag{J.4}$$

[1] PMI is based on the **mutual information** between two random variables X and Y, defined as:

$$I(X,Y) = \sum_x \sum_y P(x,y) \log_2 \frac{P(x,y)}{P(x)P(y)} \tag{J.1}$$

In a confusion of terminology, Fano used the phrase *mutual information* to refer to what we now call *pointwise mutual information* and the phrase *expectation of the mutual information* for what we now call *mutual information*

[2] Positive PMI also cleanly solves the problem of what to do with zero counts, using 0 to replace the $-\infty$ from $\log(0)$.

More formally, let's assume we have a co-occurrence matrix F with W rows (words) and C columns (contexts), where f_{ij} gives the number of times word w_i occurs with context c_j. This can be turned into a PPMI matrix where PPMI$_{ij}$ gives the PPMI value of word w_i with context c_j (which we can also express as PPMI(w_i, c_j) or PPMI($w = i, c = j$)) as follows:

$$p_{ij} = \frac{f_{ij}}{\sum_{i'=1}^{W}\sum_{j'=1}^{C} f_{i'j'}}, \quad p_{i*} = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i'=1}^{W}\sum_{j'=1}^{C} f_{i'j'}}, \quad p_{*j} = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i'=1}^{W}\sum_{j'=1}^{C} f_{i'j'}} \tag{J.5}$$

$$\text{PPMI}_{ij} = \max(\log_2 \frac{p_{ij}}{p_{i*}p_{*j}}, 0) \tag{J.6}$$

Let's see some PPMI calculations. We'll use Fig. J.2, which repeats Fig. J.1 plus all the count marginals, and let's pretend for ease of calculation that these are the only words/contexts that matter.

Here's the original figure:

	aardvark	...	**computer**	**data**	**result**	**pie**	**sugar**	...
cherry	0	...	2	8	9	442	25	...
strawberry	0	...	0	0	1	60	19	...
digital	0	...	1670	1683	85	5	4	...
information	0	...	3325	3982	378	5	13	...

Figure J.1 Co-occurrence vectors for four words in the Wikipedia corpus, showing six of the dimensions (hand-picked for pedagogical purposes). The vector for *digital* is outlined in red. Note that a real vector would have vastly more dimensions and thus be much sparser, i.e. would have zero values in most dimensions.

	computer	**data**	**result**	**pie**	**sugar**	**count(w)**
cherry	2	8	9	442	25	486
strawberry	0	0	1	60	19	80
digital	1670	1683	85	5	4	3447
information	3325	3982	378	5	13	7703
count(context)	4997	5673	473	512	61	11716

Figure J.2 Co-occurrence counts for four words in 5 contexts in the Wikipedia corpus, together with the marginals, pretending for the purpose of this calculation that no other words/contexts matter.

Thus for example we could compute PPMI(information,data), assuming we pretended that Fig. J.1 encompassed all the relevant word contexts/dimensions, as follows:

$$\begin{aligned}
P(\text{w=information, c=data}) &= \frac{3982}{11716} = .3399 \\
P(\text{w=information}) &= \frac{7703}{11716} = .6575 \\
P(\text{c=data}) &= \frac{5673}{11716} = .4842 \\
\text{PPMI(information,data)} &= \log_2(.3399/(.6575 * .4842)) = .0944
\end{aligned}$$

Fig. J.3 shows the joint probabilities computed from the counts in Fig. J.2, and Fig. J.4 shows the PPMI values. Not surprisingly, *cherry* and *strawberry* are highly associated with both *pie* and *sugar*, and *data* is mildly associated with *information*.

			p(w,context)			p(w)
	computer	**data**	**result**	**pie**	**sugar**	**p(w)**
cherry	0.0002	0.0007	0.0008	0.0377	0.0021	0.0415
strawberry	0.0000	0.0000	0.0001	0.0051	0.0016	0.0068
digital	0.1425	0.1436	0.0073	0.0004	0.0003	0.2942
information	0.2838	0.3399	0.0323	0.0004	0.0011	0.6575
p(context)	0.4265	0.4842	0.0404	0.0437	0.0052	

Figure J.3 Replacing the counts in Fig. J.1 with joint probabilities, showing the marginals in the right column and the bottom row.

	computer	**data**	**result**	**pie**	**sugar**
cherry	0	0	0	4.38	3.30
strawberry	0	0	0	4.10	5.51
digital	0.18	0.01	0	0	0
information	0.02	0.09	0.28	0	0

Figure J.4 The PPMI matrix showing the association between words and context words, computed from the counts in Fig. J.3. Note that most of the 0 PPMI values are ones that had a negative PMI; for example PMI(*cherry,computer*) = -6.7, meaning that *cherry* and *computer* co-occur on Wikipedia less often than we would expect by chance, and with PPMI we replace negative values by zero.

PMI has the problem of being biased toward infrequent events; very rare words tend to have very high PMI values. One way to reduce this bias toward low frequency events is to slightly change the computation for $P(c)$, using a different function $P_\alpha(c)$ that raises the probability of the context word to the power of α:

$$\text{PPMI}_\alpha(w,c) = \max(\log_2 \frac{P(w,c)}{P(w)P_\alpha(c)},0) \tag{J.7}$$

$$P_\alpha(c) = \frac{count(c)^\alpha}{\sum_c count(c)^\alpha} \tag{J.8}$$

Levy et al. (2015) found that a setting of $\alpha = 0.75$ improved performance of embeddings on a wide range of tasks (drawing on a similar weighting used for skip-grams described in Chapter 5. This works because raising the count to $\alpha = 0.75$ increases the probability assigned to rare contexts, and hence lowers their PMI ($P_\alpha(c) > P(c)$ when c is rare).

Another possible solution is Laplace smoothing: Before computing PMI, a small constant k (values of 0.1-3 are common) is added to each of the counts, shrinking (discounting) all the non-zero values. The larger the k, the more the non-zero counts are discounted.

Church, K. W. and P. Hanks. 1989. Word association norms, mutual information, and lexicography. *ACL*.

Church, K. W. and P. Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

Dagan, I., S. Marcus, and S. Markovitch. 1993. Contextual word similarity and estimation from sparse data. *ACL*.

Fano, R. M. 1961. *Transmission of Information: A Statistical Theory of Communications*. MIT Press.

Levy, O., Y. Goldberg, and I. Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *TACL*, 3:211–225.

Niwa, Y. and Y. Nitta. 1994. Co-occurrence vectors from corpora vs. distance vectors from dictionaries. *COLING*.