Information theoretic measures

Jean Mark Gawron

Linguistics
San Diego State University
gawron@mail.sdsu.edu
http://www.rohan.sdsu.edu/∼gawron

2026-01-23 Cs/Ling 581

Outline

Entropy

Define the **information** (magnitude) or *surprisal* of an event e to be:

$$\mathrm{I}(e) = -\log p(e)$$

Suppose $p(e_1) = .25$ and $p(e_2) = .125$, and suppose e_1 and e_2 are independent:

$$\begin{array}{lrl} & \mathrm{I}(e_1) & = -\log .25 = -\log \tfrac{1}{4} = --\log 4 = --2 = 2 \\ & \mathrm{I}(e_2) & = -\log .125 = -\log \tfrac{1}{8} = --\log 8 = --3 = 3 \\ & \mathrm{I}(e_1 \wedge e_2) & = -\log .03125 = -\log \tfrac{1}{32} = --\log 32 = --5 = 5 \\ \hline \therefore & \mathrm{I}(e_1 \wedge e_2) & = \mathrm{I}(e_1) + \mathrm{I}(e_2) \end{array}$$

Entropy of a distribution p (H(p)) is the **average** surprisal, or the **surprisal value**

$$\mathrm{H}(p) = \sum_e -p(e) \cdot \log p(e)$$

Outline

Contrasting entropies

p(H) = .25

$$\begin{aligned} H(p) &= -.25 \cdot \log .25 + -.75 \log .75 \\ &= .5 + .311 \\ &= .811 \end{aligned}$$

p(H) = .5

$$\begin{aligned} H(p) &= -.5 \cdot \log .5 + -.5 \log .5 \\ &= .5 + .5 \\ &= 1 \end{aligned}$$

Intuitions: A probability distribution captures our state of knowledge of a system

1. Entropy is related to our state of knowledge of a system. Adding information always reduces entropy. Returning to this idea when we talk about models of a system.
2. The no-information state is the uniform distribution. All outcomes are equally likely. That is also the maximum entropy distribution.
3. The maximally informed state. The probability of x is 1. The probabiity of everything else is 0. There is nothing else for us to learn. Entropy is 0.
4. Entropy is also related to compressibility.

Theorem

Shannon's Source Coding Theorem Any lossless data compression method must have an expected code length greater than or equal to the entropy of the source.

Huffman Encoding

Maximize average information in bits. No separator characters. The length of the code word for the lowest probability message m can be longer:

$$\text{Code-Word-Bit-Length}(m) \approx -\log p(m)$$

space	7	111	n	2	0010
a	4	010	s	2	1011
e	4	000	t	2	0110
f	3	1101	l	1	11001
h	2	1010	o	1	00110
i	2	1000	p	1	10011
m	2	0111	r	1	11000
			u	1	00111
			x	1	10010

Outline

Cross-entropy

For an optimal coding, Shannon's Source Coding Theorem says:

$$\mathrm{H}(X) \leq \text{Expected Code-Word-Bit-Length}(X) < \mathrm{H}(X) + 1$$

Choosing a non optimal code is analogous to modeling some data stream (e.g., language) with the wrong distribution. Using $\hat{q}$ for the model. We define cross-entropy as

$$\begin{array}{lrcl} (a) & \mathrm{H}(p, q) & = & -\sum p \log q \\ (b) & & = & \mathrm{H}(p) + \mathrm{D}_{\mathrm{kl}}(p \parallel q) \end{array}$$

The cross entropy is the expected value of the information according to q. Line (b) tells the cross entropy can be decomposed into entropy of p and the **KL-Divergence** of p and q. Since **KL-Divergence** is never negative, the cross-entropy is always greater than or equal to the entropy.

Models always add entropy

The **Kullback–Leibler divergence** (or KL-divergence) is a measure of the "distance" (or difference) between two distributions, or if p is the truth and q the model, how far the model fall short of the truth:

$$\mathrm{D_{kl}}(p \parallel q) = \sum_{x \in \chi} p(x) \log \frac{p(x)}{q(x)}$$

Not symmetric, so not a real distance (mathematically), so using the term **divergence** (divergence from) is better.

Example

Let's go back to the example of an 6-horse race R and suppose the truth is that all 6 have an equal chance of winning. $H(R) = \log_2 6 \approx 2.585$. But let's say we as bettors are misinformed that 2 of the horses are better than the others, and each has a $\frac{1}{4}$ chance. Call that distribution R'.

$$
\begin{aligned}
R &= \left(\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}\right) \\
R' &= \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}\right) \\
\text{cross-entropy}(R, R') &= \frac{1}{6}\left[(2 * 2) + (4 * 3)\right] \approx 2.666
\end{aligned}
$$

So misinformation adds entropy (confusion, surprise), even for a maximum entropy system.

Outline

"Log Perplexity"

$$
\begin{aligned}
\text{perplexity}(p) &= p_{\mathsf{C}}^{-\frac{1}{N}} \\
&= \sqrt[N]{\prod_{x \in \mathsf{C}} \frac{1}{p(x)}} \\
&= \sqrt[N]{\prod_{x \in \mathsf{C}} \frac{1}{p(x)}}
\end{aligned}
$$

$$
\log \text{ perplexity}(p) = \log(\text{perplexity}(p)) = \frac{1}{N} \sum_{x \in \mathsf{C}} -\log p(x)
$$

"Log perplexity" is cross entropy

Log perplexity is not a term we use in the field. Here's why. Let $\overline{x}$ be the type x is a token of ($p(x) = p(\overline{x})$):

$$\begin{aligned}
\text{log perplexity}(p) &= -\frac{1}{N}\sum_{x \in \mathrm{C}} \log p(x) \\
&= -\frac{1}{N}\sum_{x \in \mathrm{C}} \text{count}(\overline{x}) \log p(\overline{x}) \\
&= -\sum_{x \in \mathrm{C}} \hat{\mathrm{p}}(\overline{x}) \log p(\overline{x}) \\
&= \text{cross-entropy}(\hat{\mathrm{p}}, p)
\end{aligned}$$

We assume $\log = \log_2$. Therefore Perplexity $= 2^{\text{Cross entropy}}$.

Log likelihood and Negative Log likelihood

$$\begin{aligned} \mathrm{LL(C)} &= \sum_{x \in \mathsf{C}} \log p(x) \\ \mathrm{NLL(C)} &= -\mathrm{LL(C)} = -\sum_{x \in \mathsf{C}} \log p(x) \end{aligned}$$

NLL is a common loss function. That is, it is something we seek to minimize during training. It is what we will use whenn we build neural net language models.