

# Cross Entropy

## PA154 Jazykové modelování (2.1)

Pavel Rychlý

pary@fi.muni.cz

March 2, 2017

**Source:** Introduction to Natural Language Processing (600.465)  
Jan Hajič, CS Dept., Johns Hopkins Univ.  
[www.cs.jhu.edu/~hajic](http://www.cs.jhu.edu/~hajic)

# “Coding” Interpretation of Entropy

- The least (average) number of bits needed to encode a message (string, sequence, series, ...) (each element having being a result of a random process with some distribution  $p$ ):  $= H(p)$
- Remember various compressing algorithms?
  - ▶ they do well on data with repeating (= easily predictable = = low entropy) patterns
  - ▶ their results though have high entropy  $\Rightarrow$  compressing compressed data does nothing

## Coding: Example

- How many bits do we need for ISO Latin 1?
  - ▶  $\Rightarrow$  the trivial answer: 8
- Experience: some chars are more common, some (very) rare:
  - ▶ ... so what if we use more bits for the rare, and less bits for the frequent? (be careful: want to decode (easily)!)
    - ▶ suppose:  $p('a') = 0.3$ ,  $p('b') = 0.3$ ,  $p('c') = 0.3$ , the rest:  $p(x) \cong .0004$
    - ▶ code: 'a'  $\sim$  00, 'b'  $\sim$  01, 'c'  $\sim$  10, rest:  $11b_1b_2b_3b_4b_5b_6b_7b_8$
    - ▶ code 'acbbécbaac':

00	10	01	01	<u>1100001111</u>	10	01	00	00	10
a	c	b	b	é	c	b	a	a	c
    - ▶ number of bits used: 28 (vs. 80 using "naive" coding)
- code length  $\sim -\log(\text{probability})$

- Imagine that we produce the next letter using

$$p(l_{n+1}|l_1, \dots, l_n),$$

where  $l_1, \dots, l_n$  is the sequence of **all** the letters which had been uttered so far (i.e.  $n$  is really big!); let's call  $l_1, \dots, l_n$  the **history**  $h(h_{n+1})$ , and all histories  $H$ :

- Then compute its entropy:
  - ▶  $-\sum_{h \in H} \sum_{l \in A} p(l, h) \log_2 p(l|h)$
- Not very practical, isn't it?

- Typical case: we've got series of observations  
 $T = \{t_1, t_2, t_3, t_4, \dots, t_n\}$  (numbers, words, ...;  $t_1 \in \Omega$ );  
estimate (sample):  $\forall y \in \Omega : \tilde{p}(y) = \frac{c(y)}{|T|}$ ,  
def.  $c(y) = |\{t \in T; t = y\}|$
- ... but the true  $p$  is unknown; every sample is too small!
- Natural question: how well do we do using  $\tilde{p}$  (instead of  $p$ )?
- Idea: simulate actual  $p$  by using a different  $T$  (or rather: by using different observation we simulate the insufficiency of  $T$  vs. some other data ("random" difference))

# Cross Entropy: The Formula

- $H_{p'}(\tilde{p}) = H(p') + D(p' || \tilde{p})$

$$H_{p'}(\tilde{p}) = - \sum_{x \in \Omega} p'(x) \log_2 \tilde{p}(x)$$

- $p'$  is certainly not the true  $p$ , but we can consider it the “real world” distribution against which we test  $\tilde{p}$
- note on notation (confusing ...):  $\frac{p}{p'} \leftrightarrow \tilde{p}$ , also  $H_{T'}(p)$
- (Cross)Perplexity:  $G_{p'}(p) = G_{T'}(p) = 2^{H_{p'}(\tilde{p})}$

# Conditional Cross Entropy

- So far: “unconditional” distribution(s)  $p(x), p'(x) \dots$
- In practice: virtually always conditioning on context
- Interested in: sample space  $\Psi$ , r.v.  $Y, y \in \Psi$ ;  
context: sample space  $\Omega$ , r.v.  $X, x \in \Omega$ ;  
“our” distribution  $p(y|x)$ , test against  $p'(y, x)$ , which is taken from some independent data:

$$H_{p'}(p) = - \sum_{y \in \Psi, x \in \Omega} p'(y, x) \log_2 p(y|x)$$

# Sample Space vs. Data

- In practice, it is often inconvenient to sum over the space(s)  $\Psi, \Omega$  (especially for cross entropy!)
- Use the following formula:  
$$H_{p'}(p) = - \sum_{y \in \Psi, x \in \Omega} p'(y, x) \log_2 p(y|x) =$$
$$-1/|T'| \sum_{i=1 \dots |T'|} \log_2 p(y_i|x_i)$$
- This is in fact the normalized log probability of the “test” data:

$$H_{p'}(p) = -1/|T'| \log_2 \prod_{i=1 \dots |T'|} p(y_i|x_i)$$



# Computation Example

- $\Omega = \{a, b, \dots, z\}$ , prob. distribution (assumed/estimated from data):  $p(a) = .25$ ,  $p(b) = .5$ ,  $p(\alpha) = \frac{1}{64}$  for  $\alpha \in \{c..r\}$ ,  $= 0$  for the rest: s,t,u,v,w,x,y,z
- Data (test): barb  $p'(a) = p'(r) = .25$ ,  $p'(b) = .5$

- Sum over  $\Omega$ :

$\alpha$	<b>a</b>	<b>b</b>	<b>c</b>	<b>d</b>	<b>e</b>	<b>f</b>	<b>g</b>	<b>...</b>	<b>p</b>	<b>q</b>	<b>r</b>	<b>s</b>	<b>t</b>	<b>...</b>	<b>z</b>
$-p'(\alpha)\log_2 p(\alpha)$	<b>.5</b>	<b>+.5</b>	<b>+0</b>	<b>+0</b>	<b>+0</b>	<b>+0</b>	<b>+0</b>	<b>+0</b>	<b>+0</b>	<b>+0</b>	<b>+1.5</b>	<b>+0</b>	<b>+0</b>	<b>+0</b>	<b>+0</b>

**= 2.5**

- Sum over data:

$i/s_i$	<b>1/b</b>	<b>2/a</b>	<b>3/r</b>	<b>4/b</b>						
$-\log_2 p(s_i)$	<b>1</b>	<b>+</b>	<b>2</b>	<b>+</b>	<b>6</b>	<b>+</b>	<b>1</b>	<b>= 10</b>		

$(1/4) \times 10 = 2.5$

$\swarrow 1/|T'|$

## Cross Entropy: Some Observations

- $H(p)$  ??<, =, >??       $H_{p'}(p)$  : ALL!

- Previous example:

$p(a) = .25, p(b) = .5, p(\alpha) = \frac{1}{64}$  for  $\alpha \in \{c..r\}, = 0$  for the rest: s,t,u,v,w,x,y,z

$$H(p) = 2.5\text{bits} = H(p')(\underline{\text{barb}})$$

- Other data: probable:  $(\frac{1}{8})(6 + 6 + 6 + 1 + 2 + 1 + 6 + 6) = 4.25$

$$H(p) < 4.25\text{bits} = H(p')(\underline{\text{probable}})$$

- And finally: abba:  $(\frac{1}{4})(2 + 1 + 1 + 2) = 1.5$

$$H(p) > 1.5\text{bits} = H(p')(\underline{\text{abba}})$$

- But what about: baby  $-p'('y') \log_2 p('y') = -.25 \log_2 0 = \infty$  (??)

# Cross Entropy: Usage

- Comparing data??
  - ▶ NO! (we believe that we test on real data!)
- Rather: comparing distributions (vs. real data)
- Have (got) 2 distributions:  $p$  and  $q$  (on some  $\Omega, X$ )
  - ▶ which is better?
  - ▶ better: has lower cross-entropy (perplexity) on real data  $S$
- “Real” data:  $S$

$$H_S(p) = -1/|S| \sum_{i=1..|S|} \log_2 p(y_i|x_i) \quad ?? \quad H_S(q) = -1/|S| \sum_{i=1..|S|} \log_2 q(y_i|x_i)$$

# Comparing Distributions

- $p(\cdot)$  from previous example:

$$H_S(p) = 4.25$$

$p(a) = .25$ ,  $p(b) = .5$ ,  $p(\alpha) = \frac{1}{64}$  for  $\alpha \in \{c..r\}$ ,  $= 0$  for the rest:  $s, t, u, v, w, x, y, z$

- $q(\cdot|\cdot)$  (conditional; defined by a table):

$q(\cdot \cdot) \rightarrow$ ↓	a	b	e	l	o	p	r	other
a	0	.5	0	0	0	.125	0	0
b	1	0	0	0	1	.125	0	0
e	0	0	0	1	0	.125	0	0
l	0	.5	0	0	0	.125	0	0
o	0	0	0	0	0	.125	1	0
p	0	0	0	0	0	.125	0	1
r	0	0	0	0	0	.125	0	0
other	0	0	1	0	0	.125	0	0

ex.:  $q(o|r) = 1$

$q(r|p) = .125$

$$(1/8) (\log(p|oth.) + \log(r|p) + \log(o|r) + \log(b|o) + \log(a|b) + \log(b|a) + \log(l|b) + \log(e|l))$$

$$(1/8) ( 0 + 3 + 0 + 0 + 1 + 0 + 1 + 0 )$$

$$H_S(q) = .625$$