THE IMPACT OF CROSS-ENTROPY ON LANGUAGE MODELLING

Jie Zhao

Department for Electrical and Computer Engineering
Mississippi State University, Mississippi State, MS 39762
zhao@isip.msstate.edu

ABSTRACT

Information theory plays an important role in the field of natural language processing. Cross entropy is a distance measure from one probability distribution to another probability distribution. In language processing, language models are needed to model the uncertainty of the language. Minimum cross entropy principle has been used to construct the language model and estimate the quality of language model, translation model, etc. This paper introduces fundamentals of information theory — entropy, relative entropy, cross entropy etc. and their applications in language modelling and machine translation.

1. INTRODUCTION

Entropy is a measure of uncertainty of a random variable. Language is a kind of information source. There is a great deal of variability and uncertainty in natural language. Language modelling is the attempt to characterize the regularities, exploit the dependencies in natural language. There are basically two kinds of approaches to model the dependencies in languages [1]. One is statistical language modelling. The other is knowledge-based language modelling.

Natural language can be viewed as a stochastic process, which consists of a sequence of words. The distribution of the next word is highly dependent on the previous words. From the information theory point of view, the language has a certain inherent entropy $H(X)$. The true probability distribution of language is unknown, the model we get to model the language may be good or not. How to construct a good model and measure its quality? The perceived entropy calculated with respect to the estimated probability model is called cross entropy $H_c(X)$. and can be used to roughly estimate the true entropy. A related concept to the cross entropy is perplexity, which equals to $2^{H(X)}$.

A similar situation to the language model used in speech recognition exists in the field of machine translation. Translating from one language to another language also involves a lot of uncertainty. Translation model tries to provide the probability that a string of one language is translated to a string of another language. The cross entropy theory can be also applied into constructing and evaluating the translation model.

2. ENTROPIES

Different probabilities distributions have different uncertainties. For example, it can be easily seen that the uncertainty of probability distribution (0.5, 0.5) for a head and tail is much more than the uncertainty of the probability distribution (0.00001, 0.99999) of winning a lottery. Entropy is a measure of this uncertainty.

2.1. Entropy

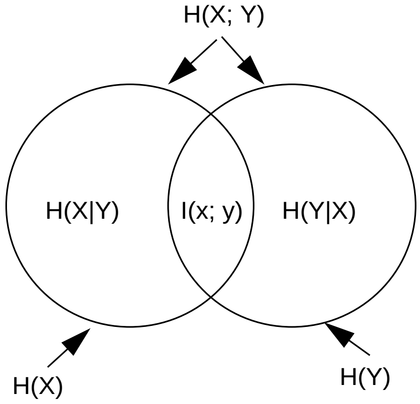
Let X be a discrete random variable with probability distribution p, its entropy is [2]:

$$\boldsymbol{H}(X) = -\sum_{x} \mathrm{p}(x) \cdot \mathbf{log}\, \mathrm{p}(x) \tag{1}$$

A property of entropy is $\boldsymbol{H}(X) \geq 0$.

The joint and condition entropy of two distributions are as follows:

$$\boldsymbol{H}(X, Y) = -\sum_{x, y} \mathrm{p}(x, y) \cdot \mathbf{log}\, \mathrm{p}(x, y) \tag{2}$$



$$
\begin{aligned}
H(X, Y) &= H(X) + H(Y|X) \\
&= H(Y) + H(X|Y) \\
I(X, Y) &= H(X) + H(Y) - H(X, Y) \\
\\
I(X, Y) &= H(X) - H(X|Y) \\
&= H(Y) - H(Y|X) \\
I(X, Y) &= I(Y, X)
\end{aligned}
$$

Figure 1: relationship between entropy and mutual information

$$
\begin{aligned}
\boldsymbol{H}(Y|X) &= \sum_{x} \mathrm{p}(x) \cdot \boldsymbol{H}(Y|(X = x)) \qquad (3) \\
&= -\sum_{x} \mathrm{p}(x) \cdot \sum_{y} [p(y|x) \cdot \log(p|x)] \\
&= -\sum_{x, y} \mathrm{p}(x, y) \cdot \mathbf{log}\, \mathrm{p}(y|x)
\end{aligned}
$$

Relative entropy (Kullback Leibler distance) between two probability distributions is defined as

$$D(p||q) = \sum_{x} \mathrm{p}(x) \cdot \log\frac{p(x)}{q(x)} \qquad (4)$$

It's a measure of the inefficiency that we assume that distribution is q, while the true distribution is p. An important property of relative entropy is that $D(p||q) \geq 0$, with zero if and only if $p = q$. Another property is $D(p||q)$ is a convex function. Also note that relative entropy is not symmetric, it is often true that

$$D(p||q) \neq D(q||p) \qquad (5)$$

A special case of relative entropy is mutual Information. It is the relative entropy between the joint distribution and the product distribution $P(x) \cdot P(y)$,

$$
\begin{aligned}
I(X, Y) &= \sum_{x, y} \mathrm{p}(x, y) \cdot \mathbf{log}\frac{\mathrm{p}(x, y)}{p(x) \cdot p(y)} \qquad (6) \\
&= \sum_{x, y} \mathrm{p}(x, y) \cdot \mathbf{log}\frac{\mathrm{p}(x|y)}{p(x)} \\
&= H(X) - H(X|Y)
\end{aligned}
$$

Mutual information is a measure of the amount of information that one random variable contains about another random variable. It is the reduction in the uncertainty of one random variable due to knowledge of the other.

All of these entropy concepts and mutual information are related to each other. The relationship between them are shown in Figure. 1.

2.2. Cross Entropy—Definition 1

There are a few definitions of cross entropy. One of them defines the cross entropy [3] the same as the relative entropy as in (4). It's a directed divergence from one distribution p to another distribution q.

Suppose $q(x)$ is a priori probability distribution, and there are a set of $\{p_i(x)\}$ (maybe infinite) that satisfy some constraints. Among these $p_i(x)$ shown in Figure 2, which is closet to the priori $q(x)$? The answer is

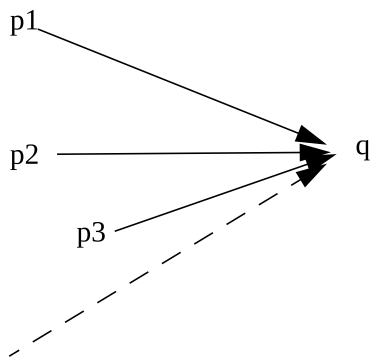


Figure 2: directed divergences from p to q

the one p_m which leads to the minimum cross entropy,

$$p_m = min(D(p_i, q)). \quad (7)$$

The above is known as the **minimum cross entropy principle**:

Given a priori distribution q, out of all probability distributions satisfying the given constraints, choose the distribution that minimizes the cross entropy of p from q. The convexity of $D(p||q)$ ensures that the cross entropy has a global minimum.

A priori distribution is the distribution what we believe the outcome should be based on our knowledge, intuition etc. The constraints come from the law of probabilities and the observations. Given the priori distribution q, minimizing

$$D(p||q) = \sum_x \mathrm{p}(x) \cdot \log\frac{p(x)}{q(x)} \quad (8)$$

subject to

$$\sum_i p_i = 1 \quad \text{and} \quad (9)$$

$$\sum_i p_i g_r(x_i) = a_r \quad (10)$$

where g_r is some function of x, the constraint is the expected value of the function of x.

Using Lagrange formulation,

$$L = -\sum_{\mathrm{i}} p_{\mathrm{i}} \cdot \mathbf{log}\frac{p_{\mathrm{i}}}{q_i} - \lambda_o\left(\sum_{\mathrm{i}} p_{\mathrm{i}} - 1\right) - \sum_r \lambda_r\left(\sum_{\mathrm{i}} p_{\mathrm{i}}\mathrm{g}_{\mathrm{ri}} - a_r\right)$$

Optimize by differentiating with respect to pi

$$\frac{dL}{dp_i} = 0 \Rightarrow -\ln\frac{p_i}{q_i} - \lambda_0 - \sum_{\mathrm{i}r} \lambda_r \mathrm{g}_{\mathrm{ri}} = 0 \quad (11)$$

we get

$$p_i = q_i \cdot \exp(-\lambda_0 - \lambda_1 g_{1i} - \ldots - \lambda_1 g_{mi}) \quad (12)$$

where λ_0 ... may be determined by using the constraints.

For example, we want to find the minimum cross-entropy probability distribution when a priori probability q is given by {0.05, 0.10, 0.15, 0.20, 0.22, 0.28} and when the mean of the six-faced die is given to be 4.5.

This problem is to minimize

$$\sum_{i=1}^{6} p_i \cdot \log\frac{p_i}{q_i} \quad (13)$$

subject to

$$\sum_{i=1}^{6} p_i = 1 \quad \text{and} \quad (14)$$

$$\sum_{i=1}^{6} i \cdot p_i = 4.5 \quad (15)$$

According to (12), we get

$$p_i = q_i \cdot \exp(-\lambda_0 - \lambda_1 i) \quad (16)$$

substitute the p_i into the constraint functions, solve them and we get the distribution

p_1 = {0.035, 0.078, 0.131, 0.192, 0.234, 0.330}

its entropy is $H(p_1) = 1.605$.

Minimizing cross entropy is an important principle in information theory. Actually, the frequently used maximum entropy principle [7] is a special case of the minimum cross entropy principle. If we don't have knowledge of the priori distribution, we will choose the $q(x)$ as uniform distribution U since it has the most uncertainty. Therefore minimizing $D(p||U)$ is equivalent to maximizing H(X).

$$D(p||U) = \sum_{x} \mathrm{p}(x) \cdot \log \frac{p(x)}{1/n} \qquad \text{x= 1,...n} \quad (17)$$
$$= \log n - \left(-\sum_{x} p(x) \cdot \log p(x)\right)$$
$$= \log n - H(X)$$

In the previous example, if we were not given the priori distribution q, this problem becomes to maximize

$$\sum_{i=1}^{6} p_i \cdot \log p_i$$

subject to the same constraints.

And the distribution is

p = {0.0543, 0.0788, 0.1142, 0.1645, 0.2378, 0.3475} with entropy $H(p_1) = 1.613$, which is greater than $H(p_1)$.

2.3. Cross Entropy—Definition 2

Another definition of cross entropy [4, 5] is

$$H_c(X) = -\sum_{x} p(x) \cdot \log q(x) \qquad (18)$$

which is the second part of (5). Since $D(p||q) \geq 0$, therefore $H_c(X) \geq H(X)$

Thereafter, the cross entropy referred in this paper refers to the second definition as in equation (18). Cross entropy is the upper bound of the entropy. The meaning is if the true distribution of a random variable is p, we know it, then we can construct a code with average description length H(p) to describe it. But now we don't know p, instead we assume that the distribution is q, so we have to use more bits on the average to describe this variable.

An directly related concept to cross entropy is **perplexity**

$$Q(X) = 2^{H_c(X)}. \qquad (19)$$

its meaning in language model will be mentioned later.

3. LANGUAGE MODELLING

Every language consists of a sequence of words. Language model is to provide the probability of next word given preceding words. In statistical language modelling [1, 7], large amount of text are used to train the language models and determine the model parameters. Another type of language models is called knowledge-based. They use linguistic knowledge to provide "yes/no" answer regarding the grammaticalily of the candidate word(s). Sometimes, they may provide a ranking of candidate words [1]. Statistical language models are more useful than the simple "yes/no" answer or even the ranking of the candidate words. And they convey more information than a simple "yes/no" answer. Therefore statistical language models are popularly used in language processing. Among the statistical language models, the dominantly used language model is the N-gram language model.

3.1. N-Gram language Model

A language model that uses the history of the $n-1$ immediately preceding words to compute the occurrence probability P of the current word is called an N-gram language model. The value of N is typically limited to 2 (bigram model) or 3 (trigram model) for feasibility. Obviously, it is not possible for an N-gram language model to estimate probabilities for all possible word pairs. Typically an N-gram lists only the most frequently occurring word pairs, and uses a backoff mechanism to compute the probability when the desired word pair is not found.

For instance, in a bigram LM, given w_i, the probability of the next word is w_j:

$$\hat{p}(w_j|w_i) = \begin{cases} p(w_j|w_i) & (w_i, w_j) \text{ exists} \\ b(w_i)p(w_j) & \text{otherwise} \end{cases} \tag{20}$$

where $b(w_i)$ is the back-off weight for the word w_i,

$p(w_i)$ is the unigram probability of the w_i

The backoff weight $b(w_i)$ is calculated to ensure that the total probability

$$\sum_j \hat{p}(w_i, w_j) = 1$$

During training the language model, discount and smoothing may be applied. In [1], an approach of combining more information source is used to generate the language model. The combined model was optimized through Estimation-Maximization (EM) algorithm to get the lowest cross entropy.

3.2. Evaluating Language Model

How do you know that one language model works better than the other? The final criterion is applying the LM into speech recognition task and getting lower word error rate. But this is very time consuming. Another way is to compare the cross entropy or perplexity of the language model over certain test data. As explained earlier, the cross entropy of the trained LM is always greater or equal than the entropy of the true distribution model (though not exist). Therefore given two models, if we can compute their cross entropy, the model with the lower entropy is closer to the true model, hence is better. Now the problem is how to compute the cross entropy since the true distribution is unknown? In [4], an approximation is given as

$$\begin{aligned} H_c(X) &= -\sum_x p(x) \cdot \log q(x) \\ &\approx -\lim_{n \to \infty} \frac{1}{n} \log q(x_1, x_2, \ldots, x_n) \end{aligned} \tag{21}$$

where $x_1, \ldots x_n$ indicates the words in the test data.

Below is the derivation of the equation (21). The entropy of a word sequence is, in fact it is the entropy rate

$$\begin{aligned} H(\underline{X}) &= \lim_{n \to \infty} \frac{1}{n} H(X_1, X_2, \ldots, X_n) \\ &= -\lim_{n \to \infty} \frac{1}{n} \sum_{\underline{x}} p(x_1, \ldots, x_n) \log p(x_1, \ldots, x_n) \end{aligned} \tag{22}$$

According to law of large numbers, we can compute temporal averages:

$$H(\underline{X}) = -\lim_{n \to \infty} \frac{1}{n} \log p(x_1, x_2, \ldots, x_n) \tag{23}$$

Similarly,

$$\begin{aligned} H_c(\underline{X}) &= \lim_{n \to \infty} \frac{1}{n} H_c(X_1, X_2, \ldots, X_n) \\ &= -\lim_{n \to \infty} \frac{1}{n} \sum_{\underline{x}} p(x_1, \ldots, x_n) \log q(x_1, \ldots, x_n) \\ &= -\lim_{n \to \infty} \frac{1}{n} \log q(x_1, x_2, \ldots, x_n) \end{aligned} \tag{24}$$

The perplexity $Q(X) = 2^{H_c(X)}$ of a language model over certain test text can be interpreted as the size of an imaginary equivalent list, in which words are equally probable. In speech recognition, the decoder will choose a word from this list when it decides which is the next word. So perplexity measures the text data complexity from the decoder point of view. Perplexity is a function of both model and text. Comparison of perplexities of several models is meaningful only when they are with respect to the same test text, same vocabulary.

3.3. Example

This example shows how to compute the cross entropy and perplexity given a back-off bigram language model and the test data.

The model was trained by CMU LM toolkits [8]. The score in the bigram file is log probability based on 10. The numbers on second column in the table were extracted from the bigram file.

	Bigram score (including back-off)	Accumulated Score
I	-1.5047 I -1.2566	-1.5047
I AM	-2.1363 I AM	-3.6410
AM A	-1.3842 AM A	-5.0252
A STUDENT	-2.8919 A STUDENT	-7.9171
STUDENT IN	-1.5866 STUDENT IN	-9.5037
IN USA	-1.9851 IN -0.6449	-9.5037-5.6413-0.6449 = -15.7899
	-5.6413 USA -0.4314	

Table 1: example of evaluating bigram over a test text.

-1.5047 I -1.2566

where -1.5047 is the unigram score of word “I”, -1.2566 is the backoff score of “I”.

-2.1363 I AM

means bigram score of “I AM” exist, -2.1363 is the bigram score.

The third column in the table is the accumulated score of these word sequences.

Perplexity is calculated as

$$Q_c(\underline{X}) = \exp\left(\frac{-1}{6}\cdot(\hat{p}(I) + \hat{p}(I, AM) + \ldots)\cdot \ln 10\right) \quad (25)$$

$$= 10^{\frac{1}{6}\times 15.79} = 428.2$$

$\hat{p}$ denotes the log probability based 10 in the bigram file.

$$H_c(\underline{X}) = \frac{1}{6}\times 15.79\times\frac{\ln 10}{\ln 2} = 8.74 bits \quad (26)$$

It’s possible that in the test set, there are words that are not in the language model. In this case, a simple way to deal with it is to get rid of this word and also deduct the total word counts. For example, if there are a sequence of w1, w2, w3, w4 and w5, but w3 is not in the bigram language model, when compute the cross entropy, we would use

$$\hat{p}(w1) + \hat{p}(w1, w2) + \hat{p}(w4) + \hat{p}(w4, w5)$$

as the accumulated probability to calculated the cross entropy and the perplexity,

4. MACHINE TRANSLATION

The task of machine translation is to translate the text of one language to the text of another language [11]. In statistical translation, for example, given a French sentence f, we see the English sentence e that maximize the $p(e|f)$.

Using Bayes law,

$$p(e|f) = \frac{p(e)\cdot p(f|e)}{p(f)} \quad (27)$$

it can be seen that maximizing $p(e|f)$ is the equivalent to maximize the $p(e)\cdot p(f|e)$. The challenges involved in the translation is

- estimating the language model of English $p(e)$

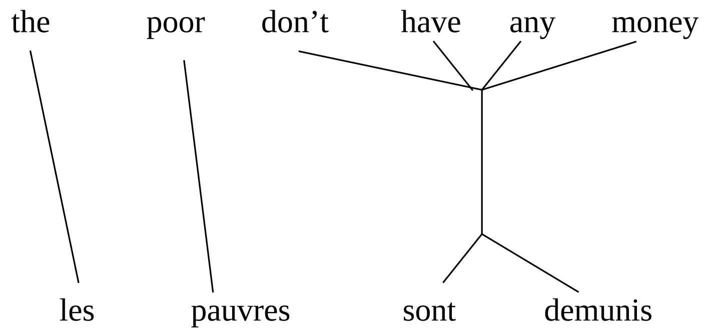


Figure 3: an example of alignment between English and French

- estimating the translation model $p(f|e)$
- effective search for the English sentence to maximize the product.

The first bullet — language modelling of English is the same as the language model we have dealt with in speech recognition.

Now we focus on the translation model $p(f|e)$. The translation model $p(f|e)$ provides the probability of translating an English sentence to a French sentence. One may wonder why not estimate $p(e|f)$ directly? If we translate from French to English, in the language model $p(e|f)$, it's important that the English sentences are well-formed, i.e they conform to some grammar constraints and are valid English sentence; but it's not important in the model $p(f|e)$, whether the French sentences are well-formed or not. Therefore for the task of translating from French to English, it would be easier to train the a better LM of $p(f|e)$ than $p(e|f)$.

Translation model needs more parameters than the N-Gram language model, we can write $p(f|e)$ in terms of the conditional probability $p(f, a|e)$ as

$$p(f, a|e) = \sum_a p(f, a|e) \qquad (28)$$

where f is a random French sentence, e is a random English sentence, a is a random alignment between them. Alignment means which word(s) in English map(s) which word(s) in French. A special case of alignment is English and French word one to one mapping. A general case is several English words map to several French words, as shown in Figure 3. So in the translation model, we need parameters such as the length of sentence, the position of the translated word, etc. There are at least 5 kinds of translation models [11, 12]. We will not talk about these models in detail in this paper.

EM (Estimation Maximization) algorithm can be used to train these translation model parameters. Each EM iteration lowers the cross entropy and perplexity. The cross entropy of a translation model over a test data can be approximated as

$$H_c(X) = -\lim_{n \to \infty}\frac{1}{n}\log q((f_1, \ldots, f_n)|(e_1, \ldots e_n)) \qquad (29)$$

where e_1 ...e_n are the given English sentences, f_1 ...f_n are the given corresponding French sentences. A good translation model will assign a high $p(f|e)$ to the given bilingual test data, hence the lower cross entropy or perplexity.

For example, given an English sentence e, and you are given $100 to bet on the translated French sentence. You can bet the $100 on only one sentence, or you can put different amount of money on different French sentences. Finally, when the result is revealed, see how much you bet on that sentence. If you bet a high probability, you win money, if you bet little amount, you lose money. If you bet 0 on that sentence, you lose everything. At each iteration of the EM algorithm, the computer plays this gambling better and better and helps you bet more money on the correct translation.

5. SUMMARY

We have described the basic concepts in information theory, such as entropy, relative entropy, mutual information. Especially, we talked about cross entropy and the principle of minimum cross entropy. Cross entropy has an important role in language modelling. We introduced the N-Gram language model, how the cross entropy is used to evaluate the language model. We also talked about translation model. During machine translation, cross entropy is also used to construct and evaluate the translation model.

The applications of cross entropy are not only restricted to the language model and translation model we mentioned in this paper. Some other applications include pattern recognition, clustering, HMM training, etc.

REFERENCES

[1] R. Rosenfeld, "Adaptive Statistical Language Modeling," Ph.D thesis, Carnegie Mellon University, Pittsburgh, PA, USA, April. 1994.

[2] T. Cover, and J. Thomas, Elements of Information Theory, John Wiley & Sons Publication, 1993.

[3] J. Kapur and H. Kesavan, Entropy Optimization Principles with Applications, Academic Press, New York, NY, USA, 1992.

[4] D. Jurafsky and J. martin, "*Speech and Language Processing — An Introduction to Speech Recognition, Natural Language Processing, and Computational Linguistics,*" Prentice Hall, New Jersey, USA, Draft of 1999.

[5] F. Jelinek, "*Statistical methods for Speech Recognition,*" Massachusetts Institute of Technology, 1997

[6] J. Deller, J. Proakis and J. Hansen, "Discrete-Time Processing of Speech Signal," MacMillan Publishing Company, 1993.

[7] P. Clarkson, and R. Rosenfeld, "Statistical Language Modelling Using the CMU-Cambridge Toolkit," proceedings ESCA Eurospeech 1997.

[8] CMU-Cambridge Language Modeling Toolkit, http://svr-www.eng.cam.ac.uk/~prc14/toolkit.html , Carnegie Mellon University, Pittsburgh, PA, USA.

[9] SRI Language Modeling toolkit, http://www.speech.sri.com/projects/srilm, SRI international Corporation, CA.

[10] http://www.isip.msstate.edu/publications/courses/ece_8993_speech/, Institute for Signal and Information Processing, Mississippi State University, May 1999.

[11] P. Brown, "the Mathematics of Machine Translation: Parameter Estimation", Computational Linguistics, vol. 19, No. 2, June 1993, pp. 263-311

[12] K. NKnight, "a Statistical Tutorial Workbook", http://www.clsp.jhu.edu/ws99/projects/mt/, JHU summer workshop, April, 1999