Notes on the KL-divergence retrieval formula and Dirichlet prior smoothing

ChengXiang Zhai

October 15, 2003

1 The KL-divergence measure

Given two probability mass functions $p(x)$ and $q(x)$, $D(p \,\|\, q)$, the Kullback-Leibler divergence (or relative entropy) between p and q is defined as

$$D(p \,\|\, q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

It is easy to show that $D(p \,\|\, q)$ is always non-negative and is zero if and only if $p = q$. Even though it is not a true distance between distributions (because it is not symmetric and does not satisfy the triangle inequality), it is still often useful to think of the KL-divergence as a "distance" between distributions [Cover and Thomas, 1991].

2 Using KL-divergence for retrieval

Suppose that a query $\mathbf{q}$ is generated by a generative model $p(\mathbf{q} \,|\, \theta_Q)$ with θ_Q denoting the parameters of the query unigram language model. Similarly, assume that a document $\mathbf{d}$ is generated by a generative model $p(\mathbf{d} \,|\, \theta_D)$ with θ_D denoting the parameters of the document unigram language model. If $\widehat{\theta}_Q$ and $\widehat{\theta}_D$ are the estimated query and document language models respectively, then, the relevance value of $\mathbf{d}$ with respect to $\mathbf{q}$ can be measured by the following *negative* KL-divergence function [Zhai and Lafferty, 2001a]:

$$-D(\widehat{\theta}_Q \,\|\, \widehat{\theta}_D) \;=\; \sum_w p(w \,|\, \widehat{\theta}_Q) \log p(w \,|\, \widehat{\theta}_D) + (-\sum_w p(w|\widehat{\theta}_Q) logp(w|\widehat{\theta}_Q))$$

Note that the second term on the right-hand side of the formula is a query-dependent constant, or more specifically, the entropy of the query model $\widehat{\theta}_Q$. It can be ignored for the purpose of ranking documents. In general, the computation of the above formula involves a sum over all the words that have a non-zero probability according to $p(w|\widehat{\theta}_Q)$. However, when $\widehat{\theta}_D$ is based on certain general smoothing method, the computation would only involve a sum over those that both have a non-zero probability according to $p(w|\widehat{\theta}_Q)$ and occur in document $\mathbf{d}$. Such a sum can be computed much more efficiently with an inverted index.

We now explain this in detail. The general smoothing scheme we assume is the following

$$p(w \,|\, \widehat{\theta}_D) \;=\; \begin{cases} p_s(w \,|\, \mathbf{d}) & \text{if word } w \text{ is seen} \\ \alpha_d \, p(w \,|\, \mathcal{C}) & \text{otherwise} \end{cases}$$

where $p_s(w \mid \mathbf{d})$ is the smoothed probability of a word seen in the document, $p(w \mid \mathcal{C})$ is the collection language model, and α_d is a coefficient controlling the probability mass assigned to unseen words, so that all probabilities sum to one. In general, α_d may depend on d. Indeed, if $p_s(w \mid \mathbf{d})$ is given, we must have

$$\alpha \;=\; \frac{1 - \sum_{w:c(w;d)>0} p_s(w \mid d)}{1 - \sum_{w:c(w;d)>0} p(w \mid \mathcal{C})}$$

Thus, individual smoothing methods essentially differ in their choice of $p_s(w \mid \mathbf{d})$.

The collection language model $p(w|\mathcal{C})$ is typically estimated by $\frac{c(w,\mathcal{C})}{\sum_{w'} c(w',\mathcal{C})}$, or a smoothed version $\frac{c(w,\mathcal{C})+1}{V+\sum_{w'} c(w',\mathcal{C})}$, where V is an estimated vocabulary size (e.g., the total number of distinct words in the collection). One advantage of the smoothed version is that it would never give a zero probability to any term, but in terms of retrieval performance, there will not be any significant difference in these two versions, since $\sum_{w'} c(w',\mathcal{C})$ is often significantly larger than V.

If can be shown that with such a smoothing scheme, the KL-divergence scoring formula is essentially:

$$\sum_{w:c(w;d)>0,\, p(w|\widehat{\theta}_Q)>0} p(w|\widehat{\theta}_Q) \log \frac{p_s(w \mid \mathbf{d})}{\alpha_d p(w \mid \mathcal{C})} + \log \alpha_d \tag{1}$$

Note that the scoring is now based on a sum over all the terms that both have a non-zero probability according to $p(w|\widehat{\theta}_Q)$ and occur in the document, i.e., all “matched” terms.

3 Using Dirichlet prior smoothing

Dirichlet prior smoothing is one particular smoothing method that follows the general smoothing scheme mentioned in the previous section. In particular,

$$p_s(w|\mathbf{d}) = \frac{c(w,d) + \mu p(w|\mathcal{C})}{|d| + \mu}$$

and

$$\alpha_d = \frac{\mu}{\mu + |d|}$$

Plugging these into equation 1, we see that with Dirichlet prior smoothing, our KL-divergence scoring formula is

$$\sum_{w:c(w;d)>0,\, p(w|\widehat{\theta}_Q)>0} p(w|\widehat{\theta}_Q) \log(1 + \frac{c(w,\mathbf{d})}{\mu p(w|\mathcal{C})}) + \log \frac{\mu}{\mu + |d|} \tag{2}$$

This is the retrieval formula that you are asked to implement in assignment 3. $p(w|\widehat{\theta}_Q)$ is passed into the function `computeWeight` as an argument. This is where the code is *different* from that in assignment 2 where the same argument carries the query term frequency. In the simplest case (i.e., initial retrieval), the probability passed in is just the normalized query term frequency (i.e., $c(w,\mathbf{q})/|\mathbf{q}|$).

4 Computing the query model $p(w|\widehat{\theta}_Q)$

You may be wondering how we can compute $p(w|\widehat{\theta}_Q)$. This is exactly where the KL-divergence retrieval method is *better* than the simple query likelihood method – we can have *different* ways of computing it! The simplest way is to estimate this probability by the maximum likelihood estimator using the query text as evidence, which gives us

$$p_{ml}(w|\widehat{\theta}_Q) = \frac{c(w, \mathbf{q})}{|q|}$$

Using this estimated value, you should see easily that the KL-divergence scoring formula is essentially the same as the query likelihood retrieval formula as presented in [Zhai and Lafferty, 2001b].

Question 1 in assignment 3 asks you to evaluate such a simple query model, which is equivalent to the query likelihood method.

A more interesting way of computing $p(w|\widehat{\theta}_Q)$ is to exploit feedback documents. Specifically, we can interpolate the simple $p_{ml}(w|\widehat{\theta}_Q)$ with a *feedback model* $p(w|\theta_F)$ estimated based on feedback documents. That is,

$$p(w|\widehat{\theta}_Q) = (1-\alpha)p_{ml}(w|\widehat{\theta}_Q) + \alpha p(w|\theta_F) \tag{3}$$

where, α is a parameter that needs to be set empirically. Please note that this α is *different* from α_d in the smoothing formula.

Of course, the next question is how to estimate $p(w|\theta_F)$? One approach is to assume the following two component mixture model for the feedback documents, where one component model is $p(w|\theta_F)$ and the other is $p(w|\mathcal{C})$, the collection language model.

$$\log p(\mathcal{F} \mid \theta_F) \quad = \quad \sum_{i=1}^{k} \sum_{w} c(w; d_i) \log((1-\lambda)p(w \mid \theta_F) + \lambda p(w \mid \mathcal{C}))$$

where, $F = \{d_1, ..., d_k\}$ is the set of feedback documents, and λ is yet another parameter that indicates the amount of "background noise" in the feedback documents, and that needs to be set empirically. Now, given λ, the feedback documents $\mathcal{F}$, and the collection language model $p(w|\mathcal{C})$, we can use the EM algorithm to compute the maximum likelihood estimate of θ_F. That is, the estimated θ_F is

$$\hat{\theta}_F = \arg\max_{\theta_F} \log p(\mathcal{F}|\theta_F)$$

The EM updating formulas are:

$$z^{(n)}(w) = \frac{(1-\lambda)p_\lambda^{(n)}(w \mid \theta_{\mathcal{F}})}{(1-\lambda)p_\lambda^{(n)}(w \mid \theta_{\mathcal{F}}) + \lambda p(w \mid \mathcal{C})}$$

$$p_\lambda^{(n+1)}(w \mid \theta_{\mathcal{F}}) = \frac{\sum_{j=1}^{k} c(w; \mathbf{d}_j) z^{(n)}(w)}{\sum_i \sum_{j=1}^{k} c(w_i; \mathbf{d}_j) z^{(n)}(w_i)}$$

Question 3 in assignment 3 asks you to complete the implementation of such an EM algorithm. All the questions after that refer to feedback, which means computing $p(w|\widehat{\theta}_Q)$ with formula 3.

References

[Cover and Thomas, 1991] Cover, T. M. and Thomas, J. A. (1991). *Elements of Information Theory*. Wiley.

[Zhai and Lafferty, 2001a] Zhai, C. and Lafferty, J. (2001a). Model-based feedback in the KL-divergence retrieval model. In *Tenth International Conference on Information and Knowledge Management (CIKM 2001)*, pages 403–410.

[Zhai and Lafferty, 2001b] Zhai, C. and Lafferty, J. (2001b). A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR'2001*, pages 334–342.