CS510: Advanced Topics in Information Retrival **Fall 2017**

Assignment 4

Kevin Li(zhuol2) *Handed In: September 28, 2017*

1. Classic Probabilistic Retrieval Model

   a. As our derivation in RSJ Model, we care about

   $$O(R=1|Q,D) = \frac{p(R=1|Q,D)}{p(R=0|Q,D)} \propto \frac{p(D|Q,R=1)}{p(D|Q,R=0)}$$

   Here we represent the document D as a set of words $w_i \in V$, where V is the vocabulary. So $w_i = 1$ represents word w_i is in the document. Then

   $$\begin{aligned}\frac{p(D|Q,R=1)}{P(D|Q,R=0)} &= \frac{\prod_{w_i \in V} p(w_i|Q,R=1)^{c(w_i,D)}}{\prod_{w_i \in V} p(w_i|Q,R=0)^{c(w_i,D)}} \\ &= \prod_{w_i \in V} (\frac{p(w_i|Q,R=1)}{p(w_i|Q,R=0)})^{c(w_i,D)}\end{aligned}$$

   Take *log* on both sides we have

   $$score(Q,D) \propto \log \frac{p(D|Q,R=1)}{p(D|Q,R=0)} = \sum_{w_i \in V} c(w_i,D) \log \frac{p(w_i|Q,R=1)}{p(w_i|Q,R=0)}$$

   There are $p(w_i|Q,R=1)$ and $p(w_i|Q,R=0)$, $2|V|$ parameters in total.

   b. We can estimate

   $$p(w_i|Q,R=0) = \frac{\sum_{j=0}^{n} c(w_i,D_j)}{|C|}$$

   c. We can estimate

   $$p(w_i|Q,R=1) = \frac{c(w_i,Q)}{|Q|}$$

   where $|D|$ is the number of words in the document

   d. We can have

   $$p(w|Q,R=1) = (1-\lambda)\frac{c(w,Q)}{|Q|} + \lambda \frac{\sum_{j=0}^{n} c(w_i,D_j)}{|C|}$$

   e. We have

   $$\begin{aligned}score(Q,D) &= \sum_{w_i \in V} c(w_i,D) \log \frac{p(w_i|Q,R=1)}{p(w_i|Q,R=0)} \\ &= \sum_{w_i \in V} c(w_i,D) \log \frac{(1-\lambda)c(w_i,Q) + \lambda|Q|p(w_i|C)}{\lambda|Q|p(w_i|C)}\end{aligned}$$

where

$$p(w_i|C) = \frac{\sum_{j=0}^{n} c(w_i, D_j)}{|C|}$$

So TF is $(1-\lambda)c(w,Q) + \lambda|Q|p(w|C)$, IDF is $\lambda|Q|p(w|C)$, document length normalization is $\lambda \log Q$

2. Language Models

a. We have

$$p(w|D) = \begin{cases} p_s(w|D), \ w \in D \\ \alpha_D p(w|REF), \ otherwise \end{cases}$$

Then

$$\frac{p_s(w|D)}{\alpha_D p(w|REF)} = \frac{(1-\lambda)p_{MLE}(w|D) + \lambda p(w|REF)}{\lambda p(w|REF)} = 1 + \frac{(1-\lambda)c(w,D)}{\lambda|D|p(w|REF)}$$

So

$$\begin{aligned} score(Q,D) &= \log p(Q|D) \\ &= \log \sum_{w \in Q \cap D} \frac{p_s(w|D)}{\alpha_D p(w|REF)} + C \end{aligned}$$

As C is a constant which can be ignore for ranking. Then

$$score(Q,D) = \sum_{w \in Q \cap D} c(w,D) \log\left(1 + \frac{(1-\lambda)c(w,D)}{\lambda|D|p(w|REF)}\right)$$

b. Query vector is

$$vec(Q) = c(\vec{w_i}, Q), w_i \in V$$

Document vector is

$$vec(D) = (1-\lambda)c(w_i, D)\vec{} + \lambda p(w_i|REF), w_i \in V$$

Similar function is

$$vec(Q)^T \cdot vec(D)$$

TF is $(1-\lambda)c(w,D) + \lambda p(w|REF)$
IDF is $\lambda|D|p(w|REF)$
Document length normalization is λ

c. For Jelinek-Mercer smoothing:

$$\begin{aligned} \log p(Q|D') &= \sum_{w \in D} c(w,D) \log \frac{k(1-\lambda)c(w,D)}{\lambda|kD|p(w|REF)} + 1 \\ &= \log p(Q|D) \end{aligned}$$

So the score always remains the same. For Dirichlet prior smoothing, we need

$$\log p(Q|D') - \log p(Q|D) \geq 0$$

Then

$$\begin{aligned}
&\sum_{w \in D} c(w,D) \log \frac{1 + kc(w,D)}{1 + c(w,D)} + |Q| \log \frac{|D| + \mu}{k|D| + \mu} \geq 0 \\
\Rightarrow\ &\log \prod_{w \in D} (\frac{1 + kc(w,D)}{1 + c(w,D)})^{c(w,Q)} + \log (\frac{|D| + \mu}{k|D| + \mu})^{|Q|} \geq 0 \\
\Rightarrow\ &\log \frac{\prod_{w \in D} (\frac{1+kc(w,D)}{1+c(w,D)})^{c(w,Q)}}{(1 + \frac{(k-1)|D|}{|D|+\mu})^{|Q|}} \geq 0 \\
\Rightarrow\ &\prod_{w \in D} (\frac{1 + kc(w,D)}{1 + c(w,D)})^{c(w,Q)} \geq (1 + \frac{(k-1)|D|}{|D| + \mu})^{|Q|}
\end{aligned}$$

So as long as our μ satisfies the above condition, our smoothing will not over-panalize.

3. KL-divergence Retrieval Function For KL-divergence retrieval function we have

$$\begin{aligned}
score(Q,D) &= \sum_{w \in V} p(w|\theta_Q) \log p(w|\theta_D) \\
&= \sum_{w \in V} \frac{c(w,Q)}{|Q|} \log p(w|\theta_D)
\end{aligned}$$

And $|Q|$ is a constant for all documents, then

$$score(Q,D) = \sum_{w \in V} c(w,Q) \log p(w|\theta_D)$$

As the query language model is to calculate $p(w|\theta_Q)$, the KL-divergence model becomes a general query language model.