# Assignment 4

Mousumi Akter

March 2020

## 1 Language Models with Smoothing

a. For Jelinek-Mercer smoothing:
$P_S(w|D) = (1-\lambda) * P_{MLE}(w|D) + \lambda * P(w|REF)$

Now,
$\frac{P_S(w|D)}{\alpha_D P(w|C)} = \frac{(1-\lambda)*P_{MLE}(w|D)+\lambda*P(w|REF)}{\lambda*P(w|REF)}$

$= 1 + \frac{(1-\lambda)*P_{MLE}(w|D)}{\lambda*P(w|REF)}$

$= 1 + \frac{(1-\lambda)*c(w,D)}{\lambda*P(w|REF)*|D|}$

Then, plugging this into the entire query likelihood retrieval formula, we get
$score(Q,D) = \sum_{w \in Q \cap D} c(w,Q) * log(1 + \frac{(1-\lambda)*c(w,D)}{\lambda*P(w|REF)*|D|})$

We ignore the $|Q|log\alpha_d$ additive term from the original score function since $\alpha_d = \lambda$ does not depend on the current document being scored.

b. The query vector is a vector of all words in the vocabulary which contains number of times a particular word present in a query.
The document vector is a vector of all words in the vocabulary which contains number of times a particular word present in a document
The similarity function is a sum over all the matched query terms.
We see very clearly the TF weighting in the numerator, which is scaled sublinearly. We also see the IDF-like weighting, which is the $p(w|REF)$ term in the denominator; the more frequent the term is in the entire collection, the more discounted the numerator will be. Finally, we can see the $|D|$ in the denominator is a form of document length normalization, since as $|D|$ grows, the overall term weight would decrease, suggesting that the impact in this case is clearly to penalize a long document.

c. For Dirichlet prior smoothing:
$P_S(w|D) = \frac{c(w,D)+\mu*p(w|C)}{|D|+\mu}$

we know that $\alpha_D = \frac{\mu}{|D|+\mu}$ . Therefore,

$$\frac{P_S(w|D)}{\alpha_D P(w|C)} = \frac{\frac{c(w,D)+\mu*p(w|C)}{|D|+\mu}}{\frac{\mu*p(w|C)}{|D|+\mu}}$$
$$= 1 + \frac{c(w,D)}{\mu*p(w|C)}$$

Then, plugging this into the entire query likelihood retrieval formula, we get

$$score(Q,D) = \sum_{w \in Q \cap D} c(w,Q) * log\{1 + \frac{c(w,D)}{\mu*p(w|C)}\} + |Q|log(\frac{\mu}{\mu+|D|})$$
$$= \sum_{w \in Q \cap D} c(w,Q) * log\{1 + \frac{c(w,D)}{\mu*p(w|C)}\} + |Q|log(\mu) - |Q|log(\mu+|D|)$$
$$\approx \sum_{w \in Q \cap D} c(w,Q) * log\{1 + \frac{c(w,D)}{\mu*p(w|C)}\} - |Q|log(\mu+|D|)$$

We ignore the $|Q|log\mu$ additive term from the original score function since it does not depend on the current document being scored.

d. The query vector is a vector of all words in the vocabulary which contains number of times a particular word present in a query.
The document vector is a vector of all words in the vocabulary which contains number of times a particular word present in a document
The similarity function is a sum over all the matched query terms along with the extra term in right side of equation.
Both TF and IDF are computed in almost the exact same way as Jelinek-Mercer scoring function. We see very clearly the TF weighting in the numerator, which is scaled sublinearly. We also see the IDF-like weighting, which is the $p(w|C)$ term in the denominator; the more frequent the term is in the entire collection, the more discounted the numerator will be. Finally, we can see the $|D|$ in the right side of the equation which is a form of document length normalization, since as $|D|$ grows, the overall term weight would decrease, suggesting that the impact in this case is clearly to penalize a long document.

e. For Jelinek-Mercer smoothing:

$$score(Q,D) = \sum_{w \in Q \cap D} c(w,Q) * log(1 + \frac{(1-\lambda)*c(w,D)}{\lambda*P(w|REF)*|D|})$$
$$score(Q,D^{'}) = \sum_{w \in Q \cap D} c(w,Q) * log(1 + \frac{(1-\lambda)*c(w,D)}{\lambda*P(w|REF)*|D|*k})$$

For Dirichlet prior smoothing:

$$score(Q,D) = \sum_{w \in Q \cap D} c(w,Q) * log\{1 + \frac{c(w,D)}{\mu*p(w|C)}\} - |Q|log(\mu+|D|)$$
$$score(Q,D^{'}) = \sum_{w \in Q \cap D} c(w,Q) * log\{1 + \frac{c(w,D)}{\mu*p(w|C)}\} - |Q|log(\mu+|D|*k)$$

None of the smoothing technique over penalize a long document due to the presence of log term which controls the penalization. However, the Dirichlet prior smoothing can capture document length normalization differently than Jelinek-Mercer smoothing. Here, we have retained the $|Q| log \alpha_D$ term since $\alpha_D$ depends on the document, namely $|D|$. If $|D|$ is large, then less extra mass is added onto the final score; if $|D|$ is small, more extra mass is added to the score, effectively rewarding a short document.