COL764 Assignment 2

Document Reranking

Prof. Srikanta B. Jagannath

Submitted by:
Mudit Soni (2017EE10463)

Implementation Details:

Language:

Python3

Files:

- freq_calc.py: Calculates term frequencies in all documents.
- prob_rerank.py: For document reranking based on PRP.
- lm_rerank.py: For document reranking based on LM.

Packages used:

- Pickle: For serializing the dictionary.
- krovetz: For krovetz stemming.

Algorithmic Details:

freq_calc.py:

1. For each term and bigram in the documents, collection term frequency and document frequency are calculated.
2. Average document length and total number are also calculated.
3. Frequencies for each term and other information are stored in a dictionary which is pickled for later access.

prob_rerank.py:

1. For each query the pre-ranked documents are retrieved from the document collection. mmap is used for efficient retrieval of documents.
2. Term frequency dictionary, generated using freq_calc.py earlier, is loaded.
3. The following formula is used to compute initial weights for each term in the document:

$$w_t = \log \frac{N}{N_t}$$

4. For each term, $n_{t,r}$ (relevant document frequency) is calculated by taking the top n_r= 20 documents based on previous rankings and top “n” terms (except those already in query) based on $n_{t,r}$*w_i values are selected.
5. After adding new terms into query, all query terms are reweighted using following formula:

$$w_t = \log \frac{(n_{t,r} + 0.5)\ (N - N_t - n_r + n_{t,r} + 0.5)}{(n_r - n_{t,r} + 0.5)\ (N_t - n_{t,r} + 0.5)}$$

6. Using the new weights, BM25 scores for all documents are calculated and documents are reranked.

$$\sum q_t \cdot \frac{f_{t,d}\,(k_1 + 1)}{k_1\ ((1 - b) + b\ (l_d / l_{avg})) + f_{t,d}} \cdot w_t$$

here, the values of hyperparameters used are:

k_1= 1.2, b= 0.75

lm_rerank.py:

1. Similar to prob_rerank.py, the documents and queries are loaded.
2. For each document, scores are calculated for each query term using the following formulas for jelinek mercer unigram and dirichlet bigram (with unigram backoff) models respectively:

$$\sum_{t \in q} q_t \cdot \log\left(1 + \frac{1 - \lambda}{\lambda} \cdot \frac{f_{t,d}}{l_d} \cdot \frac{l_C}{l_t}\right)$$

and

$$\begin{cases} \sum_{t_i, t_{i-1} \in q} q_{t_i, t_{i-1}} . \log\left(1 + \frac{f_{t_i, t_{i-1}}}{\mu} . \frac{l_C}{l_{t_i, t_{i-1}}}\right) - n . \log\left(1 + \frac{l_d}{\mu}\right) & , for\ f_{t_i, t_{i-1}} > 0 \\ \sum_{t_i \in q} q_{t_i} . \log\left(1 + \frac{f_{t_i}}{\mu} . \frac{l_C}{l_{t_i}}\right) - n . \log\left(1 + \frac{l_d}{\mu}\right) & , otherwise \end{cases}$$

3. Finally documents are re-ranked based upon the calculated scores.

Results:

Model	nDCG	MRR
Original		
PRP (n=1)		
PRP (n=2)		
PRP (n=3)		
PRP (n=4)		
PRP (n=5)		
PRP (n=6)		
PRP (n=7)		
PRP (n=8)		
PRP (n=9)		
PRP (n=10)		
LM (unigram)		
LM (bigram)		