Web Retrieval and Mining Assignment 1

B05902116 陳昱鈞

1 Vector Space Model

1.1 Calculate TF-IDF

I created a dictionary using all terms(uni-gram and bi-gram) in `inverted-file`. I also calculated the term frequency and inverse document frequency using `inverted-file`.

1.2 Processing Queries

To calculate the term frequency for queries, I used all context in each query. I gave each term in the 'title' tag double weights. Also, for all context, I gave bi-grams another double weights. I filtered out some unuseful words that don't seem to help searching, and it also turned out to be true.

1.3 Normalization

I found Okapi/BM25 normalization the best. I tried different parameters and found that $k = 4, b = 0.75$ the best.

1.4 Ranking

I calculate the cosine similarity, sort them, and output the best 100 files.

2 Rocchio Relevance Feedback

2.1 Relevant and Non-relevant documents

First, calculate the cosine similarity once. I define the top ten documents in the ranking list as relevant, and the last ten documents as non-relevant. I set $\alpha = 1, \beta = 0.8, \gamma = 0.15$.

3 Results of Experiments

3.1 MAP value under different parameters (using `ans-train.csv` and `query-train.xml`)

parameters	$k = 4, b = 0.75$	$k = 2, b = 0.75$	$k = 4, b = 0.5$	$k = 5, b = 0.75$
MAP value	0.8027	0.7724	0.7833	0.8013

With larger k and b, we got better results.

3.2 MAP value with Rocchio Relevance Feedback v.s. no Feedback (using `ans-train.csv` and `query-train.xml`)

	Rocchio Relevance Feedback	no Feedback
MAP value	0.8027	0.7885

We can conclude that with Rocchio Relevance Feedback, the performance did improve.

3.3 MAP value with Okapi normalization v.s. without Okapi normalization

	Okapi	without Okapi
MAP value	0.8027	0.6975

As you can see, Okapi normalization improved the performance a lot!

4 Discussion

In this programming assignment, I learned how to calculate TF-IDF, use normalization techniques, calculate the similarity of a document vector and a query vector, and improve performance with feedback method. The fundamentals how a search engine works are all in this assignment!