Evan Chou

Professor Sanchez Munoz

Computer Engineering 169

7 February 2021

Project 1

1. Test the performance of retrieval algorithm "RawTF" with two types of text data (i.e., raw text data and text data by stemming and removing stopwords).

a. Evaluate the results by using "../trec_eval qrel result_rawtf" and "../trec_eval qrel result_rawtf_stemmed_nostopw". Please include the results in your report. Can you tell which result is better? If one is better than the other, please provide a short analysis. Please answer what stemmer is used in the index.

result_rawtf (no removing stopwords & no stemming)	result_rawtf_stemmed_nostopw (remove stopwords & stemming)
[echou@linux10624 eval_data]$ ../trec_eval qrel result_rawtf Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 108 Interpolated Recall - Precision Averages: at 0.00 0.1760 at 0.10 0.1180 at 0.20 0.0844 at 0.30 0.0539 at 0.40 0.0396 at 0.50 0.0349 at 0.60 0.0234 at 0.70 0.0072 at 0.80 0.0072 at 0.90 0.0000 at 1.00 0.0000 Average precision (non-interpolated) for all rel docs(averaged over queries) 0.0449 Precision: At 5 docs: 0.0733 At 10 docs: 0.0833 At 15 docs: 0.0689 At 20 docs: 0.0633 At 30 docs: 0.0611 At 100 docs: 0.0360	[echou@linux10605 eval_data]$ ../trec_eval qrel result_rawtf_stemmed_nostopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 196 Interpolated Recall - Precision Averages: at 0.00 0.3991 at 0.10 0.2889 at 0.20 0.2347 at 0.30 0.2002 at 0.40 0.1186 at 0.50 0.0834 at 0.60 0.0641 at 0.70 0.0292 at 0.80 0.0292 at 0.90 0.0145 at 1.00 0.0145 Average precision (non-interpolated) for all rel docs(averaged over queries) 0.1174 Precision: At 5 docs: 0.1800 At 10 docs: 0.1433 At 15 docs: 0.1467 At 20 docs: 0.1333 At 30 docs: 0.1156 At 100 docs: 0.0653

At 200 docs: 0.0180 At 500 docs: 0.0072 At 1000 docs: 0.0036 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.0712	At 200 docs: 0.0327 At 500 docs: 0.0131 At 1000 docs: 0.0065 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.1404

From the results above, we can see that removing stopwords and stemming (text data that is preprocessed) yields better results. Although the stemmed and removed stopwords query returned the same number documents, it returned almost double the amount of relevant documents than the query without removing stopword and without stemming. The precision averages at all points of interpolated recall for rawtf_stemmed_nostopw are also significantly higher and as a result, the average precision is higher. Though removing stopwords and stemming is clearly better in this case, both results only retrieve less than half of the total relevant documents as a downside. The stemmer used in the index for these results is the porter stemmer.

b. Can you also use another stemmer and compare the results?

result_rawtf_stemmed_nostopw_krovetz (remove stopwords & stemming)

```
[echou@linux10605 eval_data]$ ../trec_eval qrel result_rawtf_stemmed_nostopw_krovetz

Queryid (Num):      30
Total number of documents over all queries
    Retrieved:    3000
    Relevant:      442
    Rel_ret:       199
Interpolated Recall - Precision Averages:
    at 0.00       0.4031
    at 0.10       0.3067
    at 0.20       0.2394
    at 0.30       0.2074
    at 0.40       0.1360
    at 0.50       0.0949
    at 0.60       0.0718
    at 0.70       0.0256
    at 0.80       0.0256
    at 0.90       0.0108
    at 1.00       0.0108
Average precision (non-interpolated) for all rel docs(averaged over queries)
                  0.1202
Precision:
  At    5 docs:   0.1800
  At   10 docs:   0.1467
  At   15 docs:   0.1533
  At   20 docs:   0.1433
  At   30 docs:   0.1144
  At  100 docs:   0.0663
```

```
  At  200 docs:   0.0332
  At  500 docs:   0.0133
  At 1000 docs:   0.0066
R-Precision (precision after R (= num_rel for a query) docs retrieved):
    Exact:        0.1444
```

The stemmer I used to compare with the porter stemmer is the krovetz stemmer. In comparison with the porter stemmer, the results from the krovetz stemmer were really similar, with only 3 more relevant documents returned. However, average precision proved to be a bit higher with krovetz stemming, possibly indicating that krovetz stemming may yield slightly better results, though a different set of documents and tokens could yield different results.

c. Evaluate the results by NOT removing the stopwords.

result_rawtf (no removing stopwords & no stemming)	result_rawtf_stemmed_stopw (no removing stopwords & stemming)

[echou@linux10624 eval_data]$ ../trec_eval qrel result_rawtf
 	
Queryid (Num): 30
Total number of documents over all queries
  Retrieved: 3000
  Relevant: 442
  Rel_ret: 108
Interpolated Recall - Precision Averages:
  at 0.00 0.1760
  at 0.10 0.1180
  at 0.20 0.0844
  at 0.30 0.0539
  at 0.40 0.0396
  at 0.50 0.0349
  at 0.60 0.0234
  at 0.70 0.0072
  at 0.80 0.0072
  at 0.90 0.0000
  at 1.00 0.0000
Average precision (non-interpolated) for all rel docs(averaged over queries)
  0.0449
Precision:
  At 5 docs: 0.0733
  At 10 docs: 0.0833
  At 15 docs: 0.0689
  At 20 docs: 0.0633
  At 30 docs: 0.0611
  At 100 docs: 0.0360
  At 200 docs: 0.0180
  At 500 docs: 0.0072
  At 1000 docs: 0.0036
R-Precision (precision after R (= num_rel for a query) docs
	[echou@linux10605 eval_data]$ ../trec_eval qrel result_rawtf_stemmed_stopw
	
	Queryid (Num): 30
	Total number of documents over all queries
	  Retrieved: 3000
	  Relevant: 442
	  Rel_ret: 121
	Interpolated Recall - Precision Averages:
	  at 0.00 0.2171
	  at 0.10 0.1443
	  at 0.20 0.1262
	  at 0.30 0.0891
	  at 0.40 0.0371
	  at 0.50 0.0300
	  at 0.60 0.0181
	  at 0.70 0.0000
	  at 0.80 0.0000
	  at 0.90 0.0000
	  at 1.00 0.0000
	Average precision (non-interpolated) for all rel docs(averaged over queries)
	  0.0545
	Precision:
	  At 5 docs: 0.0800
	  At 10 docs: 0.0533
	  At 15 docs: 0.0644
	  At 20 docs: 0.0650
	  At 30 docs: 0.0633
	  At 100 docs: 0.0403
	  At 200 docs: 0.0202
	  At 500 docs: 0.0081
	  At 1000 docs: 0.0040
	R-Precision (precision after R (= num_rel for a query) docs

retrieved): Exact: 0.0712	retrieved): Exact: 0.0625

From the results above, we can see that stemming and not removing the stopwords still yields better results than not stemming and not removing the stopwords. However, when compared to stemming and removing the stopwords, the average precision and number of returned relevant documents is actually lower. The rawtf_stemmed_nostopw results have a higher precision at all points of interpolated recall than rawtf_stemmed_stopw, revealing that not removing stopwords is less helpful than removing stopwords.

2. Implement three different retrieval algorithms (RawTFIDF, LogTFIDF, Okapi) and evaluate their performance.
Note: I used the porter stemmer for all to keep consistency.

RawTF

result_rawtf_stemmed_nostopw (remove stopwords & stemming)	result_rawtf_nostemmed_nostopw (remove stopwords & no stemming)
[echou@linux10605 eval_data]$ ../trec_eval qrel result_rawtf_stemmed_nostopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 196 Interpolated Recall - Precision Averages: at 0.00 0.3991 at 0.10 0.2889 at 0.20 0.2347 at 0.30 0.2002 at 0.40 0.1186 at 0.50 0.0834 at 0.60 0.0641 at 0.70 0.0292 at 0.80 0.0292 at 0.90 0.0145 at 1.00 0.0145 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.1174** Precision: At 5 docs: 0.1800 At 10 docs: 0.1433 At 15 docs: 0.1467 At 20 docs: 0.1333 At 30 docs: 0.1156 At 100 docs: 0.0653 At 200 docs: 0.0327	[echou@linux10605 eval_data]$ ../trec_eval qrel result_rawtf_nostemmed_nostopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 193 Interpolated Recall - Precision Averages: at 0.00 0.4982 at 0.10 0.3773 at 0.20 0.2842 at 0.30 0.2057 at 0.40 0.1428 at 0.50 0.1234 at 0.60 0.0883 at 0.70 0.0474 at 0.80 0.0391 at 0.90 0.0153 at 1.00 0.0153 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.1495** Precision: At 5 docs: 0.2067 At 10 docs: 0.1700 At 15 docs: 0.1444 At 20 docs: 0.1283 At 30 docs: 0.1189 At 100 docs: 0.0643 At 200 docs: 0.0322

At 500 docs: 0.0131 At 1000 docs: 0.0065 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.1404	At 500 docs: 0.0129 At 1000 docs: 0.0064 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.1732
result_rawtf_stemmed_stopw (no removing stopwords & stemming)	result_rawtf (no removing stopwords & no stemming)
[echou@linux10605 eval_data]$ ../trec_eval qrel result_rawtf_stemmed_stopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 121 Interpolated Recall - Precision Averages: at 0.00 0.2171 at 0.10 0.1443 at 0.20 0.1262 at 0.30 0.0891 at 0.40 0.0371 at 0.50 0.0300 at 0.60 0.0181 at 0.70 0.0000 at 0.80 0.0000 at 0.90 0.0000 at 1.00 0.0000 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.0545** Precision: At 5 docs: 0.0800 At 10 docs: 0.0533 At 15 docs: 0.0644 At 20 docs: 0.0650 At 30 docs: 0.0633 At 100 docs: 0.0403 At 200 docs: 0.0202 At 500 docs: 0.0081 At 1000 docs: 0.0040 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.0625	[echou@linux10624 eval_data]$ ../trec_eval qrel result_rawtf Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 108 Interpolated Recall - Precision Averages: at 0.00 0.1760 at 0.10 0.1180 at 0.20 0.0844 at 0.30 0.0539 at 0.40 0.0396 at 0.50 0.0349 at 0.60 0.0234 at 0.70 0.0072 at 0.80 0.0072 at 0.90 0.0000 at 1.00 0.0000 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.0449** Precision: At 5 docs: 0.0733 At 10 docs: 0.0833 At 15 docs: 0.0689 At 20 docs: 0.0633 At 30 docs: 0.0611 At 100 docs: 0.0360 At 200 docs: 0.0180 At 500 docs: 0.0072 At 1000 docs: 0.0036 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.0712

Note: Putting RawTF results above as reference only. As the directions state and the professor noted, we do not have to provide a short discussion and compare the results (advantage/disadvantage) of RawTF here because it has already been done for us.

RawTFIDF

result_rawtfidf_stemmed_nostopw (remove	result_rawtfidf_nostemmed_nostopw (remove

stopwords & stemming)	stopwords & no stemming)
[echou@linux10605 eval_data]$ ../trec_eval qrel result_rawtfidf_stemmed_nostopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 245 Interpolated Recall - Precision Averages: at 0.00 0.6691 at 0.10 0.4405 at 0.20 0.3784 at 0.30 0.2972 at 0.40 0.2462 at 0.50 0.1965 at 0.60 0.1535 at 0.70 0.1016 at 0.80 0.0825 at 0.90 0.0355 at 1.00 0.0319 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.2137** Precision: At 5 docs: 0.3067 At 10 docs: 0.2600 At 15 docs: 0.2244 At 20 docs: 0.2083 At 30 docs: 0.1744 At 100 docs: 0.0817 At 200 docs: 0.0408 At 500 docs: 0.0163 At 1000 docs: 0.0082 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.2403	[echou@linux10605 eval_data]$ ../trec_eval qrel result_rawtfidf_nostemmed_nostopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 221 Interpolated Recall - Precision Averages: at 0.00 0.6571 at 0.10 0.4869 at 0.20 0.3902 at 0.30 0.2955 at 0.40 0.2159 at 0.50 0.1865 at 0.60 0.1491 at 0.70 0.0972 at 0.80 0.0723 at 0.90 0.0310 at 1.00 0.0310 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.2170** Precision: At 5 docs: 0.2867 At 10 docs: 0.2200 At 15 docs: 0.1867 At 20 docs: 0.1750 At 30 docs: 0.1444 At 100 docs: 0.0737 At 200 docs: 0.0368 At 500 docs: 0.0147 At 1000 docs: 0.0074 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.2361
result_rawtfidf_stemmed_stopw (no removing stopwords & stemming)	result_rawtfidf (no removing stopwords & no stemming)
[echou@linux10605 eval_data]$ ../trec_eval qrel result_rawtfidf_stemmed_stopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 230 Interpolated Recall - Precision Averages: at 0.00 0.5686 at 0.10 0.4058 at 0.20 0.3385 at 0.30 0.2524	[echou@linux10605 eval_data]$ ../trec_eval qrel result_rawtfidf Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 197 Interpolated Recall - Precision Averages: at 0.00 0.6303 at 0.10 0.4414 at 0.20 0.3503 at 0.30 0.2689

at 0.40 0.1995 at 0.50 0.1505 at 0.60 0.1134 at 0.70 0.0676 at 0.80 0.0495 at 0.90 0.0136 at 1.00 0.0100 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.1764** Precision: At 5 docs: 0.2867 At 10 docs: 0.2400 At 15 docs: 0.2022 At 20 docs: 0.1817 At 30 docs: 0.1489 At 100 docs: 0.0767 At 200 docs: 0.0383 At 500 docs: 0.0153 At 1000 docs: 0.0077 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.2036	at 0.40 0.1734 at 0.50 0.1462 at 0.60 0.1153 at 0.70 0.0661 at 0.80 0.0476 at 0.90 0.0085 at 1.00 0.0085 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.1861** Precision: At 5 docs: 0.2600 At 10 docs: 0.2033 At 15 docs: 0.1711 At 20 docs: 0.1583 At 30 docs: 0.1267 At 100 docs: 0.0657 At 200 docs: 0.0328 At 500 docs: 0.0131 At 1000 docs: 0.0066 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.2147

Compared to RawTF data results, RawTFIDF performed significantly better, returning around 50% of the total number of relevant documents. Specifically, text preprocessing proved to yield better results, as the removing stopwords datas had higher average precisions (0.2137 and 0.2170 vs 0.1764 and 0.1861). However, stemming may be both beneficial and non-beneficial in RawTFIDF because of commission and omission. In the results above, it seems like stemming and removing stopwords was not as beneficial as just removing stopwords, as the average precision is slightly lower. Nonetheless, removing stopwords brings the relevant retrieved documents up to half of the total number and RawTFIDF still provides greater performance and benefit than RawTF.

LogTFIDF

result_logtfidf_stemmed_nostopw (remove stopwords & stemming)	result_logtfidf_nostemmed_nostopw (remove stopwords & no stemming)
[echou@linux10605 eval_data]$ ../trec_eval qrel result_logtfidf_stemmed_nostopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 255 Interpolated Recall - Precision Averages: at 0.00 0.7917 at 0.10 0.6549 at 0.20 0.5312	[echou@linux10605 eval_data]$ ../trec_eval qrel result_logtfidf_nostemmed_nostopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 233 Interpolated Recall - Precision Averages: at 0.00 0.7444 at 0.10 0.6458 at 0.20 0.5125

at 0.30 0.4512 at 0.40 0.4140 at 0.50 0.3221 at 0.60 0.2496 at 0.70 0.1475 at 0.80 0.1219 at 0.90 0.0643 at 1.00 0.0595 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.3241** Precision: At 5 docs: 0.4467 At 10 docs: 0.3367 At 15 docs: 0.2911 At 20 docs: 0.2550 At 30 docs: 0.2011 At 100 docs: 0.0850 At 200 docs: 0.0425 At 500 docs: 0.0170 At 1000 docs: 0.0085 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.3744	at 0.30 0.4403 at 0.40 0.3631 at 0.50 0.3054 at 0.60 0.2423 at 0.70 0.1520 at 0.80 0.1108 at 0.90 0.0511 at 1.00 0.0511 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.3119** Precision: At 5 docs: 0.4067 At 10 docs: 0.2900 At 15 docs: 0.2333 At 20 docs: 0.2117 At 30 docs: 0.1756 At 100 docs: 0.0777 At 200 docs: 0.0388 At 500 docs: 0.0155 At 1000 docs: 0.0078 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.3520
result_logtfidf_stemmed_stopw (no removing stopwords & stemming)	result_logtfidf (no removing stopwords & no stemming)
[echou@linux10605 eval_data]$ ../trec_eval qrel result_logtfidf_stemmed_stopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 235 Interpolated Recall - Precision Averages: at 0.00 0.7558 at 0.10 0.5948 at 0.20 0.5012 at 0.30 0.4207 at 0.40 0.3412 at 0.50 0.2698 at 0.60 0.1981 at 0.70 0.0893 at 0.80 0.0673 at 0.90 0.0259 at 1.00 0.0211 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.2802** Precision: At 5 docs: 0.4067 At 10 docs: 0.3133 At 15 docs: 0.2600 At 20 docs: 0.2267	[echou@linux10605 eval_data]$ ../trec_eval qrel result_logtfidf Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 206 Interpolated Recall - Precision Averages: at 0.00 0.7048 at 0.10 0.6017 at 0.20 0.4421 at 0.30 0.3915 at 0.40 0.3111 at 0.50 0.2534 at 0.60 0.1974 at 0.70 0.1011 at 0.80 0.0645 at 0.90 0.0227 at 1.00 0.0227 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.2687** Precision: At 5 docs: 0.4000 At 10 docs: 0.2600 At 15 docs: 0.2133 At 20 docs: 0.1950

At 30 docs: 0.1900 At 100 docs: 0.0783 At 200 docs: 0.0392 At 500 docs: 0.0157 At 1000 docs: 0.0078 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.2907	At 30 docs: 0.1622 At 100 docs: 0.0687 At 200 docs: 0.0343 At 500 docs: 0.0137 At 1000 docs: 0.0069 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.2857

LogTFIDF performs significantly better than RawTF and slightly better than RawTFIDF based on its results above. Compared to RawTF, its average precision across all four cases nearly triples and the number of relevant documents retrieved is further increased. Compared to RawTFIDF, it also has a higher recall and precision at all interpolated points. Also, in the case with LogTFIDF, it seems like both stemming and removing stopwords at the same time yields the best results, which indicates that there are less errors with porter stemming commissions and omissions (as opposed to it happening more frequently in RawTFIDF). The only downside is that LogTFIDF relies more on stemming and removing stopwords to be considerably more effective. That being said, LogTFIDF is clearly the best algorithm so far, as opposed to RawTF and RawTFIDF.

Okapi

result_okapi_stemmed_nostopw (remove stopwords & stemming)	result_okapi_nostemmed_nostopw (remove stopwords & no stemming)
[echou@linux10605 eval_data]$ ../trec_eval qrel result_okapi_stemmed_nostopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 286 Interpolated Recall - Precision Averages: at 0.00 0.8056 at 0.10 0.7440 at 0.20 0.6322 at 0.30 0.5556 at 0.40 0.4279 at 0.50 0.3430 at 0.60 0.2557 at 0.70 0.1752 at 0.80 0.1214 at 0.90 0.0488 at 1.00 0.0434 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.3584** Precision: At 5 docs: 0.4867	[echou@linux10605 eval_data]$ ../trec_eval qrel result_okapi_nostemmed_nostopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 255 Interpolated Recall - Precision Averages: at 0.00 0.7035 at 0.10 0.6269 at 0.20 0.5306 at 0.30 0.4494 at 0.40 0.3673 at 0.50 0.3066 at 0.60 0.2306 at 0.70 0.1661 at 0.80 0.1221 at 0.90 0.0818 at 1.00 0.0782 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.3126** Precision: At 5 docs: 0.4400

At 10 docs: 0.3967 At 15 docs: 0.3156 At 20 docs: 0.2917 At 30 docs: 0.2289 At 100 docs: 0.0953 At 200 docs: 0.0477 At 500 docs: 0.0191 At 1000 docs: 0.0095 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.3813	At 10 docs: 0.3367 At 15 docs: 0.2756 At 20 docs: 0.2383 At 30 docs: 0.1911 At 100 docs: 0.0850 At 200 docs: 0.0425 At 500 docs: 0.0170 At 1000 docs: 0.0085 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.3314
result_okapi_stemmed_stopw (no removing stopwords & stemming)	result_okapi (no removing stopwords & no stemming)
[echou@linux10605 eval_data]$ ../trec_eval qrel result_okapi_stemmed_stopw Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 291 Interpolated Recall - Precision Averages: at 0.00 0.7556 at 0.10 0.7003 at 0.20 0.6042 at 0.30 0.5030 at 0.40 0.3926 at 0.50 0.3069 at 0.60 0.2450 at 0.70 0.1689 at 0.80 0.1159 at 0.90 0.0399 at 1.00 0.0343 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.3329** Precision: At 5 docs: 0.4933 At 10 docs: 0.3900 At 15 docs: 0.3222 At 20 docs: 0.2800 At 30 docs: 0.2244 At 100 docs: 0.0970 At 200 docs: 0.0485 At 500 docs: 0.0194 At 1000 docs: 0.0097 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.3582	[echou@linux10605 eval_data]$ ../trec_eval qrel result_okapi Queryid (Num): 30 Total number of documents over all queries Retrieved: 3000 Relevant: 442 Rel_ret: 247 Interpolated Recall - Precision Averages: at 0.00 0.7113 at 0.10 0.6160 at 0.20 0.5008 at 0.30 0.4279 at 0.40 0.3383 at 0.50 0.2928 at 0.60 0.2232 at 0.70 0.1590 at 0.80 0.1227 at 0.90 0.0726 at 1.00 0.0719 **Average precision** (non-interpolated) for all rel docs(averaged over queries) **0.3004** Precision: At 5 docs: 0.4267 At 10 docs: 0.3167 At 15 docs: 0.2644 At 20 docs: 0.2300 At 30 docs: 0.1833 At 100 docs: 0.0823 At 200 docs: 0.0412 At 500 docs: 0.0165 At 1000 docs: 0.0082 R-Precision (precision after R (= num_rel for a query) docs retrieved): Exact: 0.3136

Out of all the retrieval algorithms, Okapi is clearly the one that yields the best results. Compared to all of the other algorithms, the results above show that it has the higher average

precision for all four cases and the precision averages at all points of interpolated recall are the highest we have seen so far. The number of relevant documents retrieved is also around 60% of the total number of relevant documents, which tells us that this is a pretty good retrieval algorithm. Similar to all of the others, specifically LogTFIDF, it benefits from stemming and removing stopwords. However, compared to LogTFIDF and RawTFIDF, the other average precision results are significantly lower than stemming and stopword removal. For instance, okapi_stemmed_stopw has an average precision that is just slightly lower than the average precision in okapi_stemmed_nostopw. The only downside to Okapi is that it requires the most information about documents, including document length and average document length. As a result, if we desire less calculations or don't have enough overhead for extra data on documents, RawTFIDF and LogTFIDF would be our second choices.

In general, Okapi is the best choice, as it performs the best on all levels and shows the best numbers for recall and precision.