Lucene Search Engine

CS7IS3 Information Retrieval and Web Search

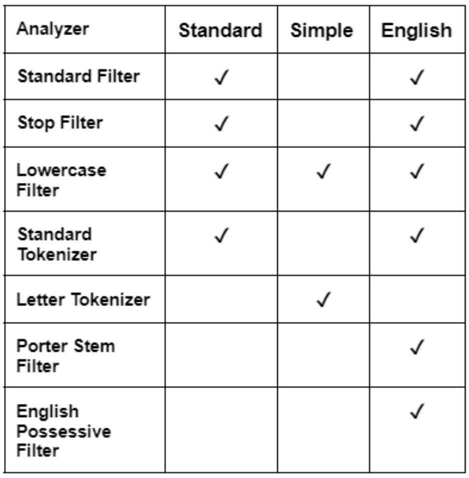
Gwen McArdle
18322248

Background

Analyzer

Analyzers handle the preprocessing for indexing and querying. I selected some of the popular analysers to test; StandardAnalyzer, SimpleAnalyzer and EnglishAnalyzer. ("Guide to Lucene Analyzers" 2021) Analyzers are made up of a combination of tokenizers and filters which perform the preprocessing techniques. The below table illustrates which tokenizers and filters each analyzer uses.

("StandardAnalyzer (Lucene 7.3.1 API)", n.d.), ("SimpleAnalyzer (Lucene 6.4.0 API)", n.d.), ("EnglishAnalyzer (Lucene 8.1.1 API)", n.d.)



Analyzer	Standard	Simple	English
Standard Filter	✓		✓
Stop Filter	✓		✓
Lowercase Filter	✓	✓	✓
Standard Tokenizer	✓		✓
Letter Tokenizer		✓	
Porter Stem Filter			✓
English Possessive Filter			✓

Standard filter - Normalises tokens. ("StandardFilter (Lucene 7.0.0 API)", n.d.)
Stop filter - Removes stop words from a token stream. ("StopFilter (Lucene 8.0.0 API)", n.d.)
Lowercase filter - Normalises token text to lower case. ("LowerCaseFilter (Lucene 8.0.0 API)", n.d.)
Standard tokenizer - Splits text into a stream of tokens based on grammar. ("StandardTokenizer (Lucene 6.6.0 API)", n.d.)
Letter tokenizer - Tokenizer that divides text at non-letters. ("LetterTokenizer (Lucene 7.3.1 API)", n.d.)
No tokenizer - The keyword analyzer does not use a tokenizer. The entire stream is treated as a single token.
PorterStem Filter - Transforms the token stream as per the Porter stemming algorithm. ("PorterStemFilter (Lucene 7.4.0 API)", n.d.)
EnglishPossessive Filter - TokenFilter that removes possessives from words. ("EnglishPossessiveFilter", n.d.)

Similarities

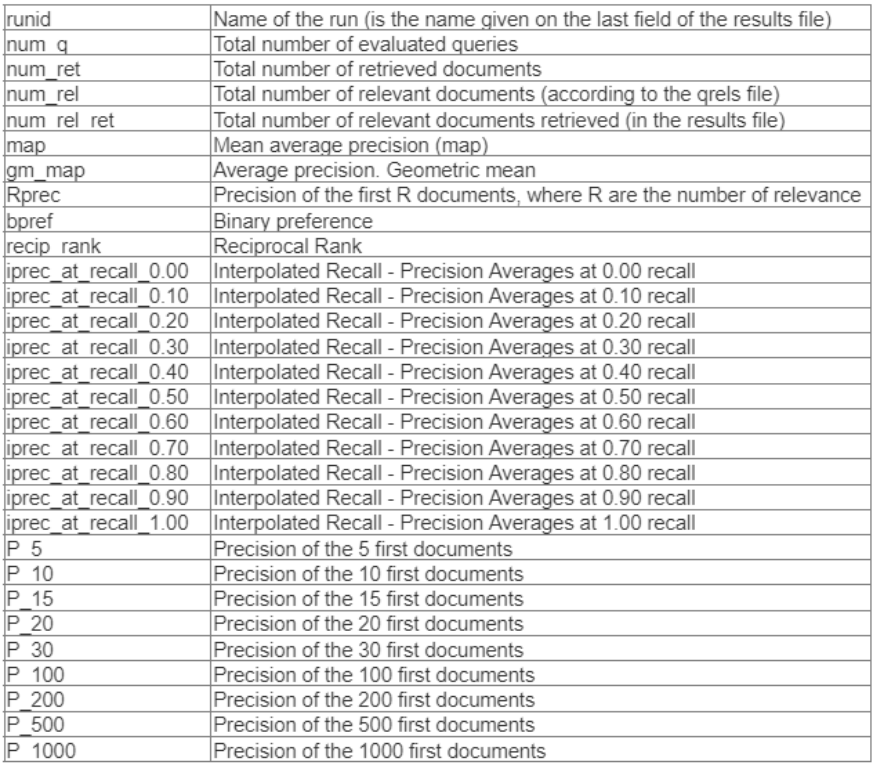
Similarities handle the scoring of the documents in response to query. The same similarity should be applied at both index and query time. ("Uses of Class org.apache.lucene.search.similarities.Similarity (Lucene 6.2.1 API)", n.d.)
The ClassicSimilarity is the "default scoring implementation which encodes norm values as a single byte, before being stored." ("Uses of Class org.apache.lucene.search.similarities.Similarity (Lucene 6.2.1 API)", n.d.) It is a subclass of TFIDFSimilarity and is therefore a Vector Space Model based similarity. ("TFIDFSimilarity (Lucene 8.2.0 API)", n.d.)
The BooleanSimilarity is a "simple similarity that gives terms a score that is equal to their query boost." ("BooleanSimilarity (Lucene 8.2.0 API)", n.d.) If I were to select this similarity for a task, I would do further experimentation with different boost factors, than I did for the purpose of this assignment.
The BM25Similarity gives a score based on the algorithm below. It takes two arguments; k1 - which controls nonlinear term frequency normalisation, and b - which controls to what degree document length normalises tf values. ("BM25Similarity (Lucene 8.2.0 API)", n.d.)

$$score(q,d) = \sum_{i=1}^{|q|} idf(q_i) \cdot \frac{tf(q_i,d)\cdot(k_1+1)}{tf(q_i,d)+k_1\cdot(1-b+b\cdot\frac{|d|}{avgdl}}$$

(Athens University of Economics and Business, n.d.)

TRECeval

The trec_eval is a tool used to evaluate rankings. It takes two inputs; a document that lists the relevance judgements for each query, and a document that lists the results of the search engine under examination, and their corresponding scores. The tool returns a series of results, as explained in this table from Rafael Glater's article. ("Learn how to use trec_eval to evaluate your information retrieval system" 2016)



runid	Name of the run (is the name given on the last field of the results file)
num_q	Total number of evaluated queries
num_ret	Total number of retrieved documents
num_rel	Total number of relevant documents (according to the qrels file)
num_rel_ret	Total number of relevant documents retrieved (in the results file)
map	Mean average precision (map)
gm_map	Average precision. Geometric mean
Rprec	Precision of the first R documents, where R are the number of relevance
bpref	Binary preference
recip_rank	Reciprocal Rank
iprec_at_recall_0.00	Interpolated Recall - Precision Averages at 0.00 recall
iprec_at_recall_0.10	Interpolated Recall - Precision Averages at 0.10 recall
iprec_at_recall_0.20	Interpolated Recall - Precision Averages at 0.20 recall
iprec_at_recall_0.30	Interpolated Recall - Precision Averages at 0.30 recall
iprec_at_recall_0.40	Interpolated Recall - Precision Averages at 0.40 recall
iprec_at_recall_0.50	Interpolated Recall - Precision Averages at 0.50 recall
iprec_at_recall_0.60	Interpolated Recall - Precision Averages at 0.60 recall
iprec_at_recall_0.70	Interpolated Recall - Precision Averages at 0.70 recall
iprec_at_recall_0.80	Interpolated Recall - Precision Averages at 0.80 recall
iprec_at_recall_0.90	Interpolated Recall - Precision Averages at 0.90 recall
iprec_at_recall_1.00	Interpolated Recall - Precision Averages at 1.00 recall
P_5	Precision of the 5 first documents
P_10	Precision of the 10 first documents
P_15	Precision of the 15 first documents
P_20	Precision of the 20 first documents
P_30	Precision of the 30 first documents
P_100	Precision of the 100 first documents
P_200	Precision of the 200 first documents
P_500	Precision of the 500 first documents
P_1000	Precision of the 1000 first documents

Results and Discussion

Boost

Boosted Value	None	Title	Author	Bibliography	Content
Mean Average Precision	0.2047	0.2732	0.1005	0.0520	0.2047

I experimented with boosting each field one at a time. From these trials it became clear that boosting Title was the only field that had a positive effect on the results. I tried a few different boost factors for Title and it had an insignificant effect.

Analyzers and Similarity

Below is a sample of some of the TRECeval results that I believe communicate the pros and cons of the analyzers and similarities. The full results can be found in the tables in the appendix of this report. This table is in order of Mean Average Precision (map).

Analyzer	Similarity	map	num_rel - num_rel_ret	iprec_at_recall _0.50	iprec_at_recall _1.0	P_5	P_30
English	BM25(k1=1.2, b=0.5)	0.3246	366	0.3197	0.0732	0.3618	0.1283
English	BM25(k1=0.6, b=0.75)	0.3241	366	0.3198	0.0734	0.3627	0.1283
English	BM25(k1=1.2, b=0.75)	0.323	366	0.3137	0.077	0.3556	0.1277
English	classic	0.3194	366	0.3079	0.0798	0.3556	0.4459
English	boolean	0.3012	366	0.2973	0.0601	0.3618	0.4264
Simple	BM25(k1=1.2, b=0.75)	0.2775	172	0.2554	0.0588	0.3236	0.1132
Simple	BM25(k1=0.6, b=0.75)	0.2739	172	0.2602	0.0558	0.3173	0.112
Standard	BM25(k1=1.2, b=0.75)	0.2735	181	0.248	0.0585	0.3209	0.1135
Simple	BM25(k1=1.2, b=0.5)	0.2723	172	0.2573	0.0551	0.3164	0.112
Standard	BM25(k1=0.6, b=0.75)	0.2715	181	0.2576	0.0555	0.3138	0.1121
Standard	BM25(k1=1.2, b=0.5)	0.2677	181	0.2528	0.0547	0.3111	0.1119
Simple	classic	0.2549	172	0.2336	0.0555	0.3236	0.3598
Standard	classic	0.2526	181	0.2302	0.0552	0.3209	0.3569
Standard	boolean	0.1813	181	0.1617	0.0279	0.3111	0.2464
Simple	boolean	0.181	172	0.1612	0.0268	0.3164	0.2466

This table clearly communicates that the EnglishAnalyzer is the most suitable for the Cran dataset. It performs best across most precision measures as well as the measures that are dependent on recall. However, it yields fewer relevant results than the other analyzers. This makes sense as it has the more types of relevant preprocessing (e.g. stemming and possessives removing), than the other analyzers used.

The SimpleAnalyzer seems to perform slightly better than the StandardAnalyzer, but the similarities they are paired with has a significant effect on the order of how well they perform. The SimpleAnalyzer is closest to the ideal in terms of relevant results.

The table also demonstrates that the BM25 similarities are more performant for this dataset than the ClassicSimilarity, which in turn is more performant than the BooleanSimilarity. However, this does not hold over all of the returned documents, when looking at a larger number of the top documents, I began to see the ClassicSimilarity and BooleanSimilarity become more precise than the BM25 similarities.

I ran three different versions of the BM25. The default version is when k=1.2 and b=0.75. I tried adjusting each input and there is no clear winner. Different parameters were more suitable to be combined with different analyzers.

References

Athens University of Economics and Business. n.d. “The BM25 similarity function.” http://ipl.cs.aueb.gr/stougiannis/.

“BM25Similarity (Lucene 8.2.0 API).” n.d. Apache Lucene. Accessed October 21, 2022. https://lucene.apache.org/core/8_2_0/core/org/apache/lucene/search/similarities/BM25Similarity.html.

“BooleanSimilarity (Lucene 8.2.0 API).” n.d. Apache Lucene. Accessed October 21, 2022. https://lucene.apache.org/core/8_2_0/core/org/apache/lucene/search/similarities/BooleanSimilarity.html.

“EnglishAnalyzer (Lucene 8.1.1 API).” n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/8_1_1/analyzers-common/org/apache/lucene/analysis/en/EnglishAnalyzer.html.

“EnglishPossessiveFilter.” n.d. Lucene 8.0.0 API. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/8_3_0/analyzers-common/org/apache/lucene/analysis/en/EnglishPossessiveFilter.html.

“Guide to Lucene Analyzers.” 2021. Baeldung. https://www.baeldung.com/lucene-analyzers.

“KeywordAnalyzer (Lucene 6.6.0 API).” n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/6_6_0/analyzers-common/org/apache/lucene/analysis/core/KeywordAnalyzer.html.

“Learn how to use trec_eval to evaluate your information retrieval system.” 2016. Rafael Glater. http://www.rafaelglater.com/en/post/learn-how-to-use-trec_eval-to-evaluate-your-information-retrieval-system.

“LetterTokenizer (Lucene 7.3.1 API).” n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/7_3_1/analyzers-common/org/apache/lucene/analysis/core/LetterTokenizer.html.

“LowerCaseFilter (Lucene 8.0.0 API).” n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/8_0_0/analyzers-common/org/apache/lucene/analysis/core/LowerCaseFilter.html.

“PorterStemFilter (Lucene 7.4.0 API).” n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/6_4_1/analyzers-common/org/apache/lucene/analysis/en/PorterStemFilter.html.

"SimpleAnalyzer (Lucene 6.4.0 API)." n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/6_4_0/analyzers-common/org/apache/lucene/analysis/core/SimpleAnalyzer.html.

"StandardAnalyzer (Lucene 7.3.1 API)." n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/7_3_1/core/org/apache/lucene/analysis/standard/StandardAnalyzer.html.

"StandardFilter (Lucene 7.0.0 API)." n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/7_3_1/core/org/apache/lucene/analysis/standard/StandardFilter.html.

"StandardTokenizer (Lucene 6.6.0 API)." n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/6_6_0/core/org/apache/lucene/analysis/standard/StandardTokenizer.html.

"StopAnalyzer (Lucene 8.0.0 API)." n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/8_0_0/analyzers-common/org/apache/lucene/analysis/core/StopAnalyzer.html.

"StopFilter (Lucene 8.0.0 API)." n.d. Apache Lucene. Accessed October 20, 2022. https://lucene.apache.org/core/8_0_0/analyzers-common/org/apache/lucene/analysis/core/StopFilter.html.

"TFIDFSimilarity (Lucene 8.2.0 API)." n.d. Apache Lucene. Accessed October 21, 2022. https://lucene.apache.org/core/8_2_0/core/org/apache/lucene/search/similarities/TFIDFSimilarity.html.

"Uses of Class org.apache.lucene.search.similarities.Similarity (Lucene 6.2.1 API)." n.d. Apache Lucene. Accessed October 21, 2022. https://lucene.apache.org/core/6_2_1/core/org/apache/lucene/search/similarities/class-use/Similarity.html.

Appendix

Boost selection process

Boosted Value	None	Title	Author	Bibliography	Content
num_q	225	225	225	225	225
num_ret	231705	231705	231705	231705	231705
num_rel	1837	1837	1837	1837	1837
num_rel_ret	1656	1656	1656	1656	1656
map	0.2047	0.2732	0.1005	0.0520	0.2047
gm_map	0.1158	0.1720	0.0539	0.0345	0.1158
Rprec	0.2126	0.2731	0.1201	0.0297	0.2126
bpref	0.9098	0.9098	0.9098	0.9098	0.9098
recip_rank	0.4992	0.6699	0.1909	0.0815	0.4992
iprec_at_recall_0.00	0.5203	0.6840	0.2401	0.1170	0.5203
iprec_at_recall_0.10	0.4734	0.6294	0.2261	0.1074	0.4734
iprec_at_recall_0.20	0.3835	0.5046	0.1979	0.0980	0.3835
iprec_at_recall_0.30	0.2956	0.3857	0.1600	0.0831	0.2956
iprec_at_recall_0.40	0.2162	0.2937	0.1224	0.0695	0.2162
iprec_at_recall_0.50	0.1833	0.2500	0.1030	0.0626	0.1833
iprec_at_recall_0.60	0.1310	0.1725	0.0770	0.0516	0.1310
iprec_at_recall_0.70	0.1061	0.1366	0.0620	0.0418	0.1061
iprec_at_recall_0.80	0.0712	0.0914	0.0451	0.0314	0.0712
iprec_at_recall_0.90	0.0522	0.0653	0.0317	0.0221	0.0522
iprec_at_recall_1.00	0.0455	0.0574	0.0253	0.0172	0.0455
P_5	0.2329	0.3191	0.1289	0.0267	0.2329
P_10	0.1720	0.2204	0.1133	0.0258	0.1720
P_15	0.1407	0.1730	0.0963	0.0293	0.1407
P_20	0.1240	0.1462	0.0849	0.0300	0.1240
P_30	0.1006	0.1141	0.0695	0.0293	0.1006
P_100	0.0440	0.0460	0.0387	0.0388	0.0440
P_200	0.0265	0.0276	0.0244	0.0261	0.0265
P_500	0.0129	0.0130	0.0121	0.0129	0.0129
P_1000	0.0072	0.0072	0.0072	0.0072	0.0072

Analyzer and similarity combination selection

Analyzer	**Standard**	**Simple**	**English**	**Standard**	**Simple**	**English**
Similarity	**Classic**	**Classic**	**Classic**	**Boolean**	**Boolean**	**Boolean**
num_q	225	225	225	225	225	225
num_ret	231705	234016	76615	231705	234016	76615
num_rel	1837	1837	1837	1837	1837	1837
num_rel_ret	1656	1665	1471	1656	1665	1471
map	0.2526	0.2549	0.3194	0.1813	0.181	0.3012
gm_map	0.1509	0.1528	0.2074	0.0792	0.0791	0.1835
Rprec	0.2504	0.2562	0.3072	0.1887	0.186	0.2974
bpref	0.9098	0.9167	0.8187	0.9098	0.9167	0.8187
recip_rank	0.619	0.6174	0.7156	0.4988	0.4985	0.7005
iprec_at_recall_0.00	0.6399	0.6365	0.7389	0.5139	0.5129	0.7191
iprec_at_recall_0.10	0.5964	0.5948	0.6972	0.4598	0.46	0.677
iprec_at_recall_0.20	0.4734	0.4755	0.5679	0.3428	0.3439	0.5397
iprec_at_recall_0.30	0.3569	0.3598	0.4459	0.2464	0.2466	0.4264
iprec_at_recall_0.40	0.2662	0.2692	0.3669	0.1837	0.1835	0.3486
iprec_at_recall_0.50	0.2302	0.2336	0.3079	0.1617	0.1612	0.2973
iprec_at_recall_0.60	0.1632	0.1674	0.222	0.1139	0.1137	0.2165
iprec_at_recall_0.70	0.1216	0.1261	0.1743	0.0844	0.0849	0.1618
iprec_at_recall_0.80	0.0821	0.0818	0.1191	0.0482	0.0469	0.1035
iprec_at_recall_0.90	0.0624	0.0628	0.0869	0.0324	0.0312	0.0678
iprec_at_recall_1.00	0.0552	0.0555	0.798	0.0279	0.0268	0.0601
P_5	0.3004	0.3013	0.352	0.2133	0.2133	0.3369
P_10	0.2031	0.2044	0.2462	0.1418	0.1413	0.2253
P_15	0.1653	0.1668	0.192	0.1147	0.1135	0.1775
P_20	0.1389	0.1396	0.1644	0.0996	0.0976	0.148
P_30	0.1087	0.1089	0.1268	0.081	0.0807	0.1159
P_100	0.0439	0.0442	0.0513	0.0338	0.0339	0.0489
P_200	0.0262	0.0264	0.0297	0.021	0.021	0.029
P_500	0.0126	0.0127	0.013	0.0113	0.0113	0.013
P_1000	0.0072	0.0072	0.0065	0.007	0.007	0.0065

Analyzer	Standard	Simple	English	Standard	Simple	English
Similarity	BM25 (k1=1.2, b=0.75)	BM25 (k1=1.2, b=0.75)	BM25 (k1=1.2, b=0.75)	BM25 (k1=1.2, b=0.5)	BM25 (k1=1.2, b=0.5)	BM25 (k1=1.2, b=0.5)
num_q	225	225	225	225	225	225
num_ret	231705	234016	76615	231705	234016	76615
num_rel	1837	1837	1837	1837	1837	1837
num_rel_ret	1656	1665	1471	1656	1665	1471
map	0.2735	0.2775	0.323	0.2677	0.2723	0.3246
gm_map	0.1721	0.1754	0.2098	0.1653	0.1691	0.2086
Rprec	0.2718	0.2762	0.3105	0.2685	0.271	0.3151
bpref	0.9098	0.9167	0.8187	0.9298	0.9167	0.8187
recip_rank	0.6743	0.6748	0.7277	0.6497	0.6567	0.7227
iprec_at_recall_0.00	0.6881	0.6897	0.7488	0.6674	0.6745	0.7411
iprec_at_recall_0.10	0.6307	0.6384	0.6995	0.6163	0.6248	0.6979
iprec_at_recall_0.20	0.5074	0.5123	0.5681	0.41936	0.5019	0.5676
iprec_at_recall_0.30	0.3844	0.3889	0.447	0.377	0.3862	0.4475
iprec_at_recall_0.40	0.2919	0.297	0.3702	0.2892	0.2962	0.3745
iprec_at_recall_0.50	0.248	0.2554	0.3137	0.2528	0.2573	0.3197
iprec_at_recall_0.60	0.1723	0.18	0.2302	0.1725	0.1769	0.2373
iprec_at_recall_0.70	0.1361	0.1407	0.1832	0.1362	0.1395	0.1908
iprec_at_recall_0.80	0.0919	0.0921	0.1196	0.0878	0.0878	0.119
iprec_at_recall_0.90	0.0663	0.0666	0.0842	0.0625	0.063	0.0807
iprec_at_recall_1.00	0.0585	0.0588	0.077	0.0547	0.0551	0.0732
P_5	0.3209	0.3236	0.3556	0.3111	0.3164	0.3618
P_10	0.2178	0.2227	0.2516	0.2164	0.2178	0.2471
P_15	0.1739	0.1748	0.1938	0.1716	0.1721	0.1929
P_20	0.1469	0.1471	0.1642	0.1464	0.1471	0.1642
P_30	0.1135	0.1132	0.1277	0.1119	0.112	0.1283
P_100	0.0459	0.0462	0.0512	0.046	0.046	0.0512
P_200	0.0276	0.0276	0.0297	0.0274	0.0274	0.0297
P_500	0.013	0.013	0.013	0.0129	0.013	0.013
P_1000	0.0072	0.0072	0.0065	0.0072	0.0072	0.0065

Analyzer	Standard	Simple	English
Similarity	BM25(k1=0.6, b=0.75)	BM25(k1=0.6, b=0.75)	BM25(k1=0.6, b=0.75)
num_q	225	225	225
num_ret	231705	234016	76615
num_rel	1837	1837	1837
num_rel_ret	1656	1665	1471
map	0.2715	0.2739	0.3241
gm_map	0.1685	0.1713	0.2088
Rprec	0.2677	0.2729	0.3165
bpref	0.9098	0.9167	0.8187
recip_rank	0.6625	0.6623	0.7223
iprec_at_recall_0.00	0.6781	0.6798	0.7417
iprec_at_recall_0.10	0.625	0.6307	0.6988
iprec_at_recall_0.20	0.4975	0.5032	0.5679
iprec_at_recall_0.30	0.3805	0.3856	0.4483
iprec_at_recall_0.40	0.2946	0.2985	0.3756
iprec_at_recall_0.50	0.2576	0.2602	0.3198
iprec_at_recall_0.60	0.1748	0.1774	0.2352
iprec_at_recall_0.70	0.1391	0.1408	0.1892
iprec_at_recall_0.80	0.089	0.089	0.1189
iprec_at_recall_0.90	0.0634	0.0639	0.0809
iprec_at_recall_1.00	0.0555	0.0558	0.0734
P_5	0.3138	0.3173	0.3627
P_10	0.2169	2164	0.2484
P_15	0.173	0.1745	0.1938
P_20	0.1467	0.1471	0.1638
P_30	0.1121	0.112	0.1283
P_100	0.0464	0.0463	0.0511
P_200	0.0274	0.0275	0.0297
P_500	0.013	0.013	0.013
P_1000	0.0072	0.0072	0.0065