IR - PROJECT 3

PART 1: Describe how to implement each model in Solr and provide screenshots on your key implementation and results to demonstrate that you have successfully implemented them.

Implementation of the IR Models:

We have declared all the similarities globally in schema.xml and the below default results are for 20 queries which were given to train our systems and rows = 20.

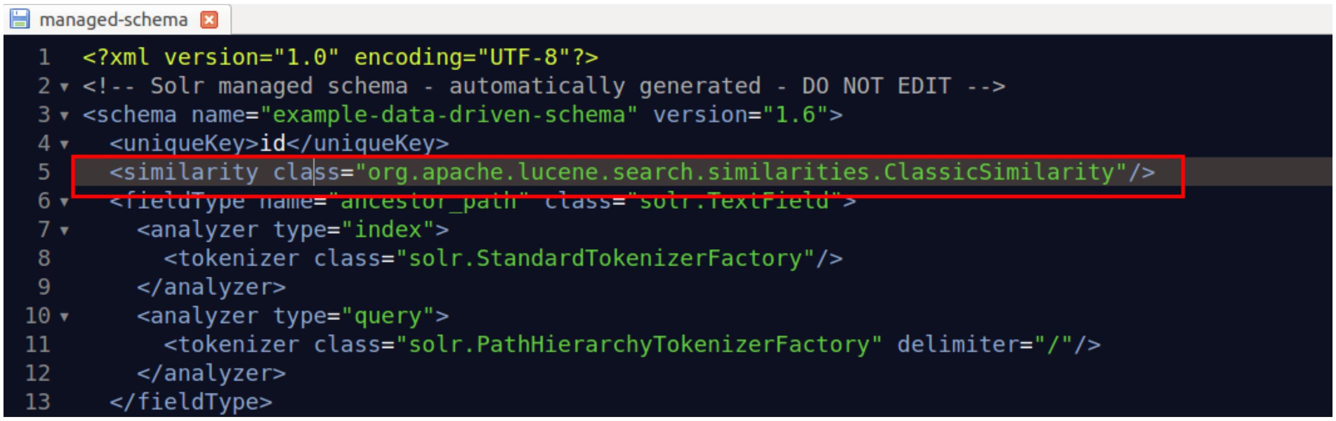
i. Vector Space Model

We implemented ClasicSimilarity which is dependent on Vector Space model but as this this did not give any parameters to modify we also implemented SweetSpotSimilarityFactory which is a subclass of Classic Similarity. The implementation details for both are as below.

(a) Classic Similarity

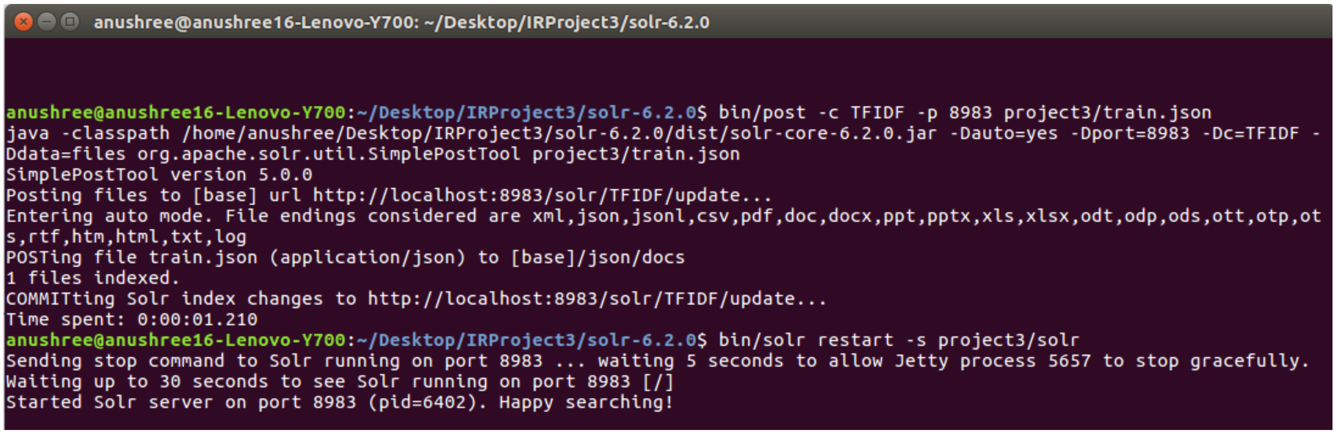
URL:https://lucene.apache.org/core/5_5_0/core/org/apache/lucene/search/similarities/ClassicSimilarity.html

Modifications in Schema.xml: Classic Similarity does not have any parameters and is declared as below.



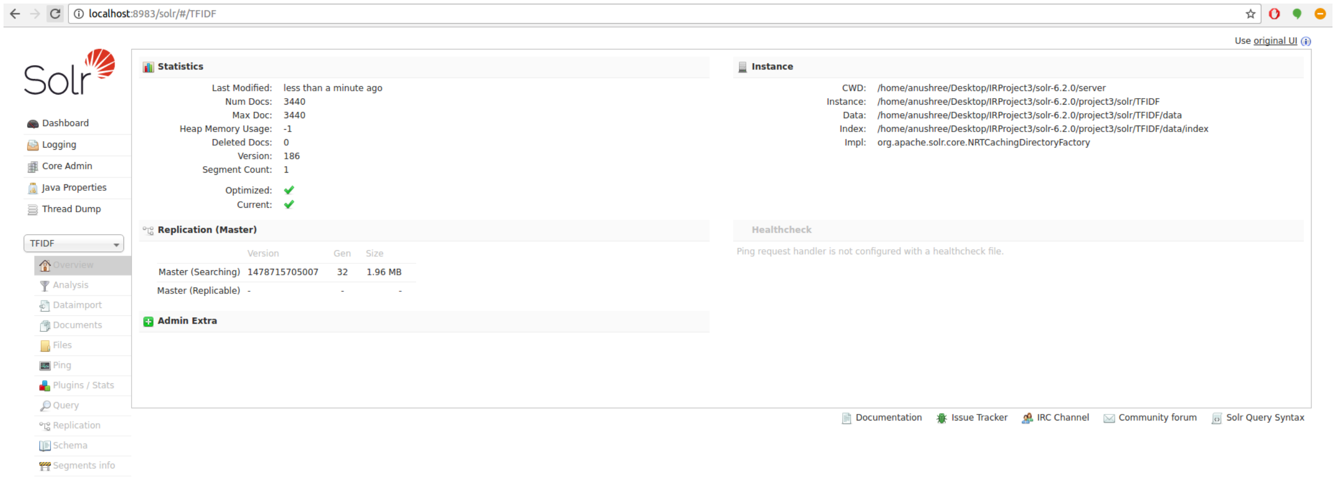
```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Solr managed schema - automatically generated - DO NOT EDIT -->
<schema name="example-data-driven-schema" version="1.6">
  <uniqueKey>id</uniqueKey>
  <similarity class="org.apache.lucene.search.similarities.ClassicSimilarity"/>
  <fieldType name="ancestor_path" class="solr.TextField">
    <analyzer type="index">
      <tokenizer class="solr.StandardTokenizerFactory"/>
    </analyzer>
    <analyzer type="query">
      <tokenizer class="solr.PathHierarchyTokenizerFactory" delimiter="/"/>
    </analyzer>
  </fieldType>
```

Indexing data to Solr Query Results: We index the train.json with the newly changed schema.xml in Solr using terminal commands.


anushree@anushree16-Lenovo-Y700: ~/Desktop/IRProject3/solr-6.2.0

```
anushree@anushree16-Lenovo-Y700:~/Desktop/IRProject3/solr-6.2.0$ bin/post -c TFIDF -p 8983 project3/train.json
java -classpath /home/anushree/Desktop/IRProject3/solr-6.2.0/dist/solr-core-6.2.0.jar -Dauto=yes -Dport=8983 -Dc=TFIDF -Ddata=files org.apache.solr.util.SimplePostTool project3/train.json
SimplePostTool version 5.0.0
Posting files to [base] url http://localhost:8983/solr/TFIDF/update...
Entering auto mode. File endings considered are xml,json,jsonl,csv,pdf,doc,docx,ppt,pptx,xls,xlsx,odt,odp,ods,ott,otp,ots,rtf,htm,html,txt,log
POSTing file train.json (application/json) to [base]/json/docs
1 files indexed.
COMMITting Solr index changes to http://localhost:8983/solr/TFIDF/update...
Time spent: 0:00:01.210
anushree@anushree16-Lenovo-Y700:~/Desktop/IRProject3/solr-6.2.0$ bin/solr restart -s project3/solr
Sending stop command to Solr running on port 8983 ... waiting 5 seconds to allow Jetty process 5657 to stop gracefully.
Waiting up to 30 seconds to see Solr running on port 8983 [/]
Started Solr server on port 8983 (pid=6402). Happy searching!
```

Verifying if the documents are indexed on frontend.



Statistics

Last Modified: less than a minute ago
Num Docs: 3440
Max Doc: 3440
Heap Memory Usage: -1
Deleted Docs: 0
Version: 186
Segment Count: 1
Optimized: ✓
Current: ✓

Replication (Master)

	Version	Gen	Size
Master (Searching)	1478715705007	32	1.96 MB
Master (Replicable)	-	-	-

Admin Extra

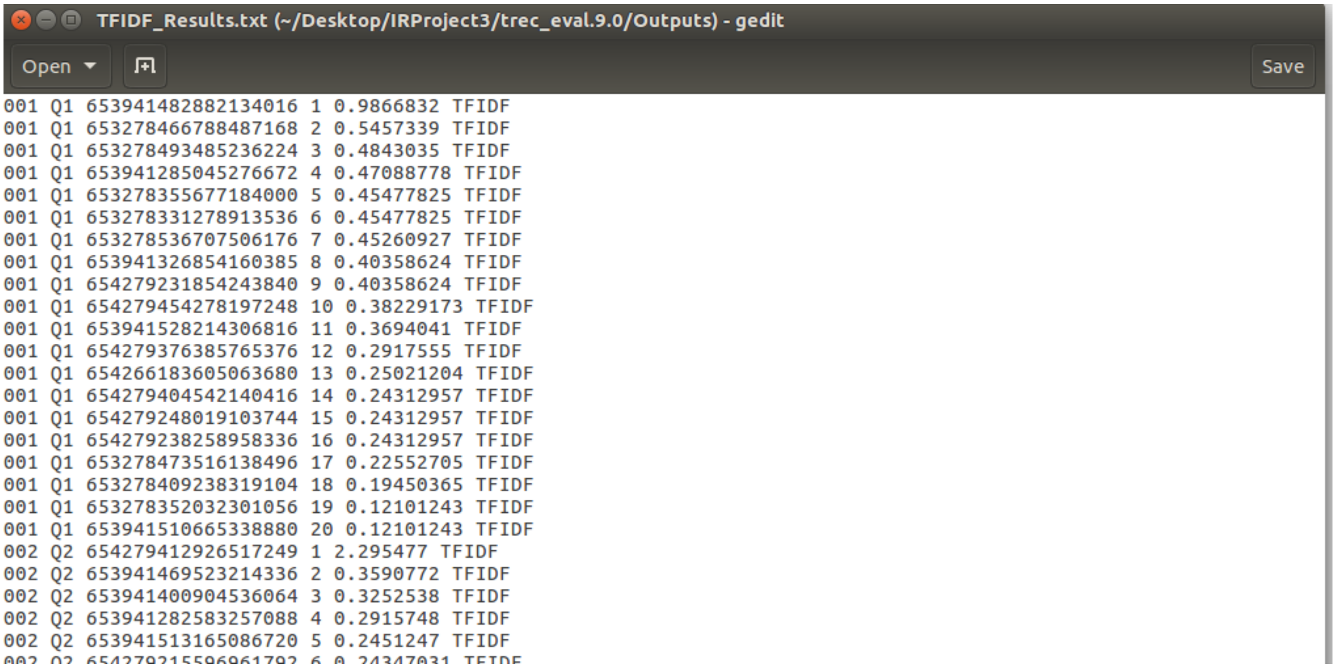
Instance

CWD: /home/anushree/Desktop/IRProject3/solr-6.2.0/server
Instance: /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/TFIDF
Data: /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/TFIDF/data
Index: /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/TFIDF/data/index
Impl: org.apache.solr.core.NRTCachingDirectoryFactory

Healthcheck

Ping request handler is not configured with a healthcheck file.

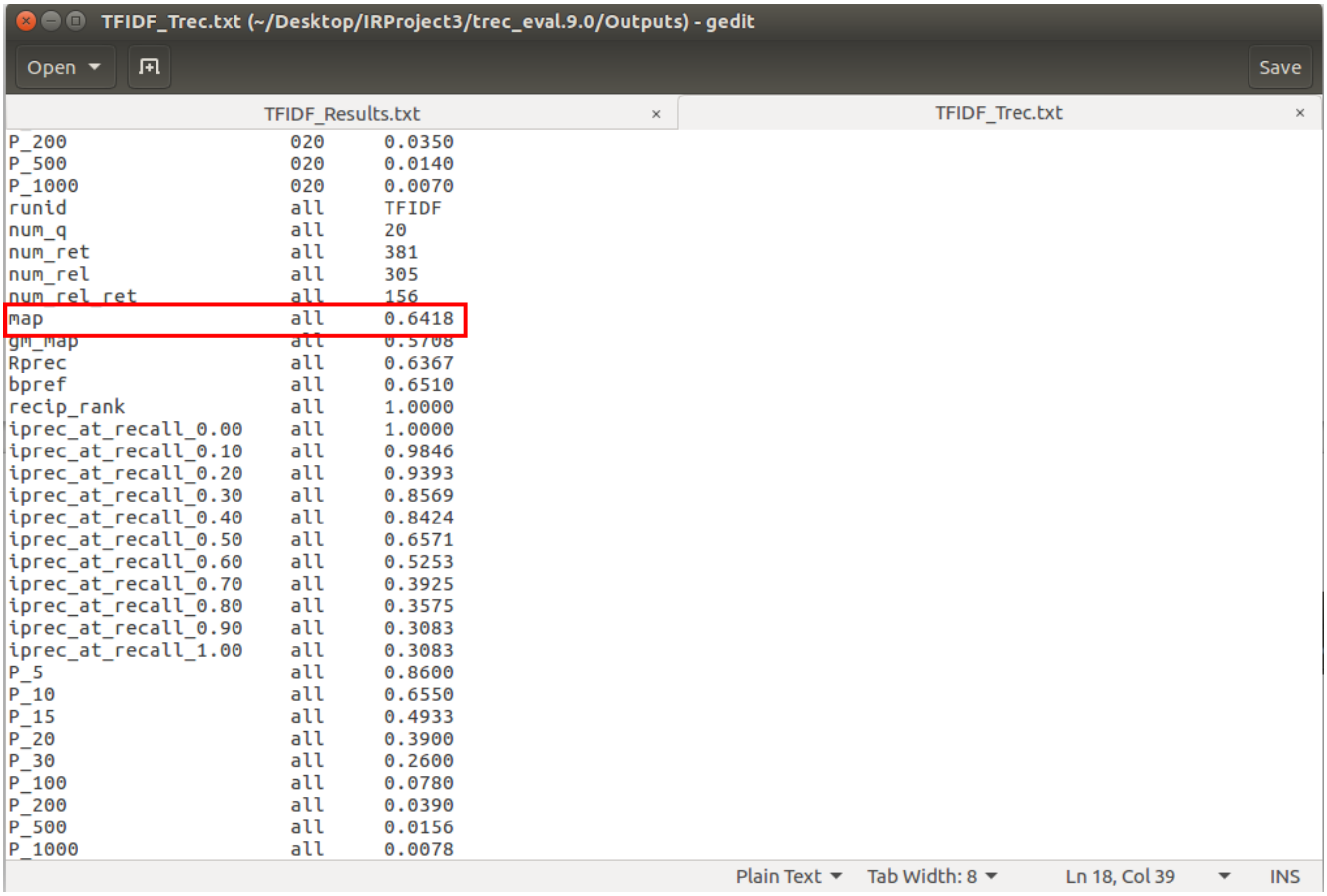
Query Results: Next we run the json_to_trec.py to get the output in format of trec input.


TFIDF_Results.txt (~/Desktop/IRProject3/trec_eval.9.0/Outputs) - gedit
Open
Save

```
001 Q1 653941482882134016 1 0.9866832 TFIDF
001 Q1 653278466788487168 2 0.5457339 TFIDF
001 Q1 653278493485236224 3 0.4843035 TFIDF
001 Q1 653941285045276672 4 0.47088778 TFIDF
001 Q1 653278355677184000 5 0.45477825 TFIDF
001 Q1 653278331278913536 6 0.45477825 TFIDF
001 Q1 653278536707506176 7 0.45260927 TFIDF
001 Q1 653941326854160385 8 0.40358624 TFIDF
001 Q1 654279231854243840 9 0.40358624 TFIDF
001 Q1 654279454278197248 10 0.38229173 TFIDF
001 Q1 653941528214306816 11 0.3694041 TFIDF
001 Q1 654279376385765376 12 0.2917555 TFIDF
001 Q1 654266183605063680 13 0.25021204 TFIDF
001 Q1 654279404542140416 14 0.24312957 TFIDF
001 Q1 654279248019103744 15 0.24312957 TFIDF
001 Q1 654279238258958336 16 0.24312957 TFIDF
001 Q1 653278473516138496 17 0.22552705 TFIDF
001 Q1 653278409238319104 18 0.19450365 TFIDF
001 Q1 653278352032301056 19 0.12101243 TFIDF
001 Q1 653941510665338880 20 0.12101243 TFIDF
002 Q2 654279412926517249 1 2.295477 TFIDF
002 Q2 653941469523214336 2 0.3590772 TFIDF
002 Q2 653941400904536064 3 0.3252538 TFIDF
002 Q2 653941282583257088 4 0.2915748 TFIDF
002 Q2 653941513165086720 5 0.2451247 TFIDF
002 Q2 654279215596961792 6 0.24347031 TFIDF
```

MAP Value via TREC_eval: We run trec_eval on our query results via command line using the below command.

```
./trec_eval -q -c -M3440 Outputs/qrel.txt Outputs/TFIDF_Results.txt > Outputs/TFIDF_Trec.txt
```

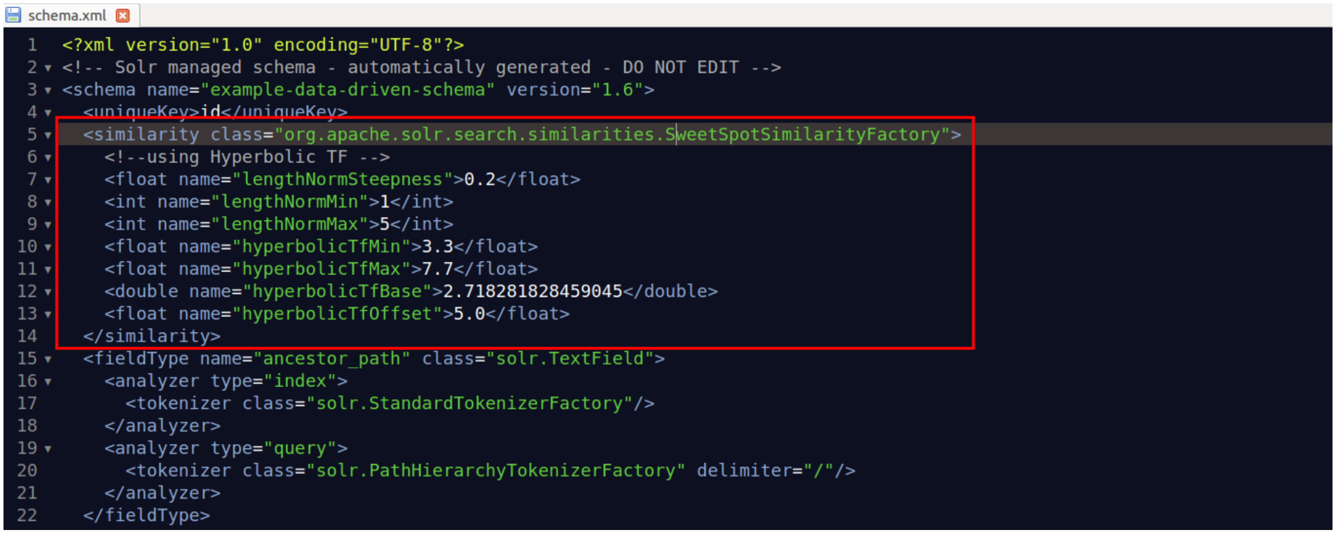


```
P_200                   020     0.0350
P_500                   020     0.0140
P_1000                  020     0.0070
runid                   all     TFIDF
num_q                   all     20
num_ret                 all     381
num_rel                 all     305
num_rel_ret             all     156
map                     all     0.6418
gm_map                  all     0.5708
Rprec                   all     0.6367
bpref                   all     0.6510
recip_rank              all     1.0000
iprec_at_recall_0.00    all     1.0000
iprec_at_recall_0.10    all     0.9846
iprec_at_recall_0.20    all     0.9393
iprec_at_recall_0.30    all     0.8569
iprec_at_recall_0.40    all     0.8424
iprec_at_recall_0.50    all     0.6571
iprec_at_recall_0.60    all     0.5253
iprec_at_recall_0.70    all     0.3925
iprec_at_recall_0.80    all     0.3575
iprec_at_recall_0.90    all     0.3083
iprec_at_recall_1.00    all     0.3083
P_5                     all     0.8600
P_10                    all     0.6550
P_15                    all     0.4933
P_20                    all     0.3900
P_30                    all     0.2600
P_100                   all     0.0780
P_200                   all     0.0390
P_500                   all     0.0156
P_1000                  all     0.0078
```

(b) SweetSpotSimilarityFactory

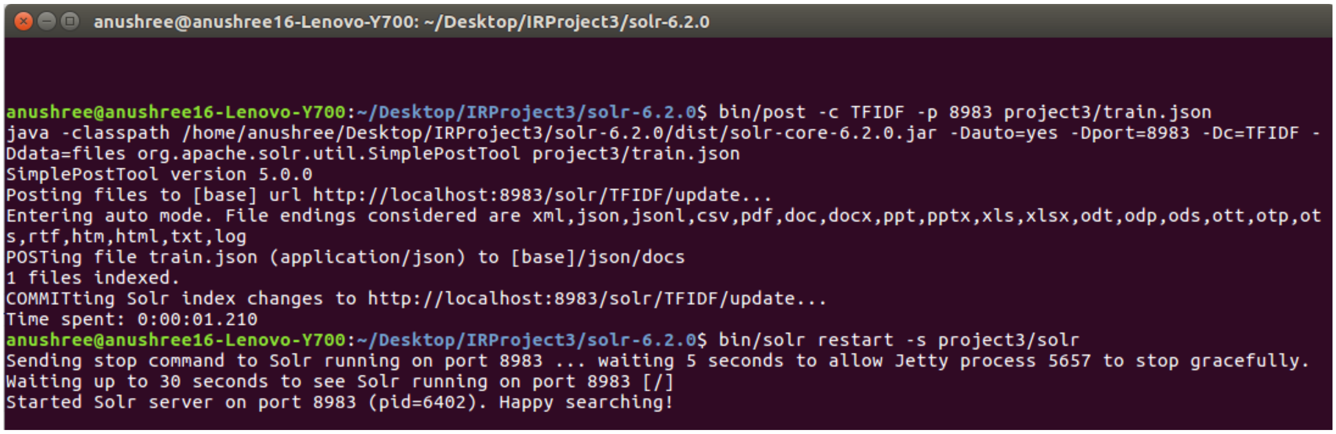
URL: http://lucene.apache.org/solr/6_0_0/solr-core/org/apache/solr/search/similarities/SweetSpotSimilarityFactory.html

Modifications in Schema.xml: We have used SweetSpot similarity using Hyperbolic TF with default values in schema.xml.



```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Solr managed schema - automatically generated - DO NOT EDIT -->
<schema name="example-data-driven-schema" version="1.6">
  <uniqueKey>id</uniqueKey>
  <similarity class="org.apache.solr.search.similarities.SweetSpotSimilarityFactory">
    <!--using Hyperbolic TF -->
    <float name="lengthNormSteepness">0.2</float>
    <int name="lengthNormMin">1</int>
    <int name="lengthNormMax">5</int>
    <float name="hyperbolicTfMin">3.3</float>
    <float name="hyperbolicTfMax">7.7</float>
    <double name="hyperbolicTfBase">2.718281828459045</double>
    <float name="hyperbolicTfOffset">5.0</float>
  </similarity>
  <fieldType name="ancestor_path" class="solr.TextField">
    <analyzer type="index">
      <tokenizer class="solr.StandardTokenizerFactory"/>
    </analyzer>
    <analyzer type="query">
      <tokenizer class="solr.PathHierarchyTokenizerFactory" delimiter="/"/>
    </analyzer>
  </fieldType>
```

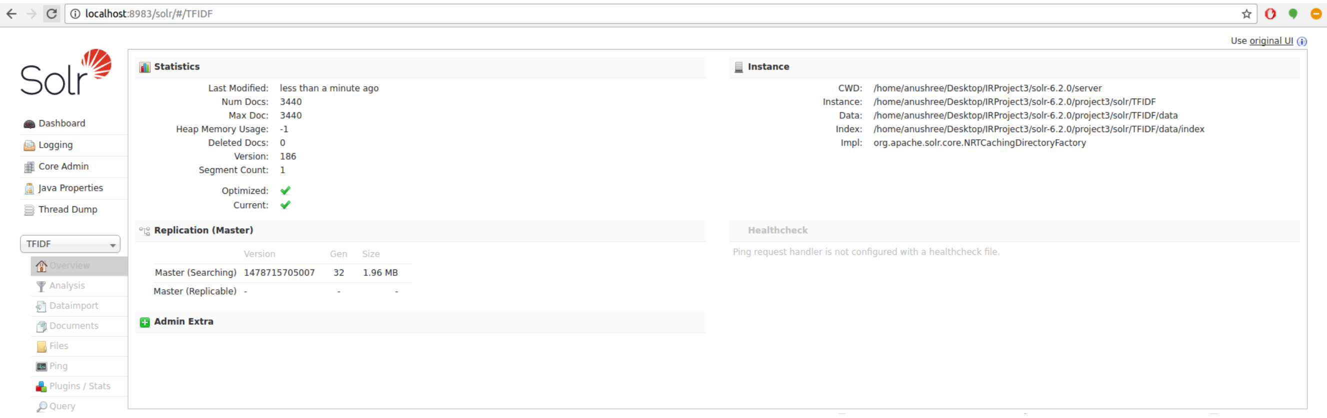
Indexing data to Solr Query Results: We index the train.json with the newly changed schema.xml in Solr using terminal commands.



anushree@anushree16-Lenovo-Y700: ~/Desktop/IRProject3/solr-6.2.0

```
anushree@anushree16-Lenovo-Y700:~/Desktop/IRProject3/solr-6.2.0$ bin/post -c TFIDF -p 8983 project3/train.json
java -classpath /home/anushree/Desktop/IRProject3/solr-6.2.0/dist/solr-core-6.2.0.jar -Dauto=yes -Dport=8983 -Dc=TFIDF -
Ddata=files org.apache.solr.util.SimplePostTool project3/train.json
SimplePostTool version 5.0.0
Posting files to [base] url http://localhost:8983/solr/TFIDF/update...
Entering auto mode. File endings considered are xml,json,jsonl,csv,pdf,doc,docx,ppt,pptx,xls,xlsx,odt,odp,ods,ott,otp,ot
s,rtf,htm,html,txt,log
POSTing file train.json (application/json) to [base]/json/docs
1 files indexed.
COMMITting Solr index changes to http://localhost:8983/solr/TFIDF/update...
Time spent: 0:00:01.210
anushree@anushree16-Lenovo-Y700:~/Desktop/IRProject3/solr-6.2.0$ bin/solr restart -s project3/solr
Sending stop command to Solr running on port 8983 ... waiting 5 seconds to allow Jetty process 5657 to stop gracefully.
Waiting up to 30 seconds to see Solr running on port 8983 [/]
Started Solr server on port 8983 (pid=6402). Happy searching!
```

Verifying if the documents are indexed on frontend.



Statistics

```
     Last Modified:  less than a minute ago
         Num Docs:   3440
          Max Doc:   3440
Heap Memory Usage:   -1
     Deleted Docs:   0
          Version:   186
    Segment Count:   1
        Optimized:   ✓
          Current:   ✓
```

Replication (Master)

	Version	Gen	Size
Master (Searching)	1478715705007	32	1.96 MB
Master (Replicable)	-	-	-

Admin Extra

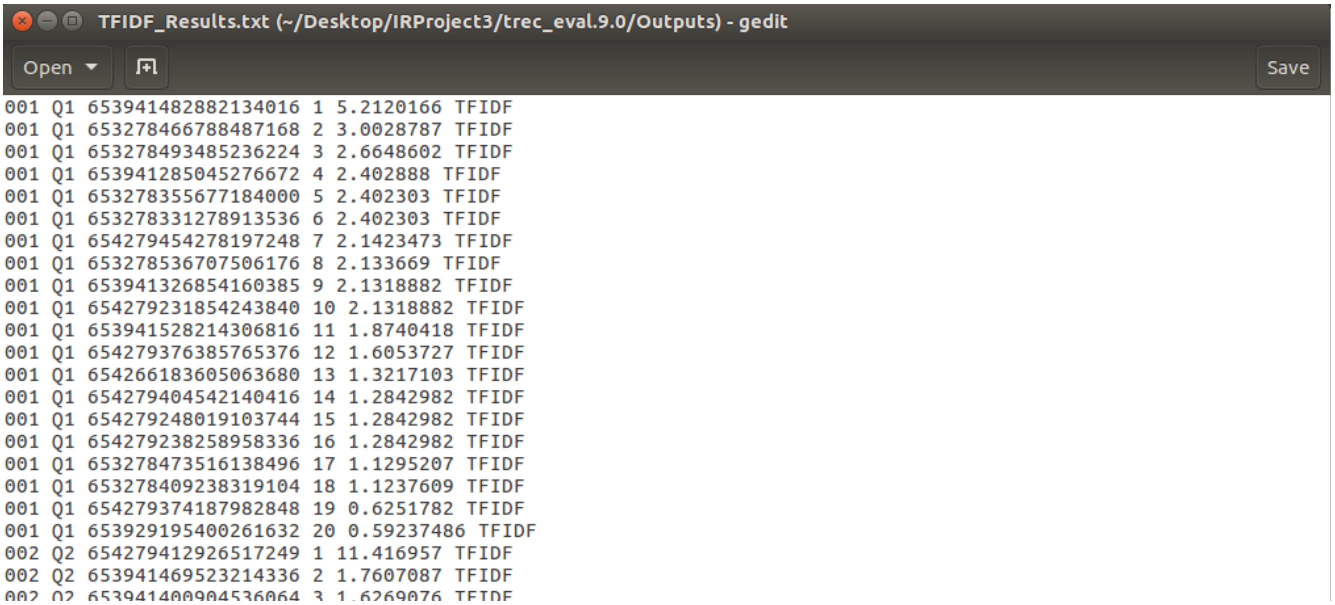
Instance

```
      CWD:  /home/anushree/Desktop/IRProject3/solr-6.2.0/server
 Instance:  /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/TFIDF
     Data:  /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/TFIDF/data
    Index:  /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/TFIDF/data/index
     Impl:  org.apache.solr.core.NRTCachingDirectoryFactory
```

Healthcheck

Ping request handler is not configured with a healthcheck file.

Query Results: Next we run the json_to_trec.py to get the output in format of trec input.

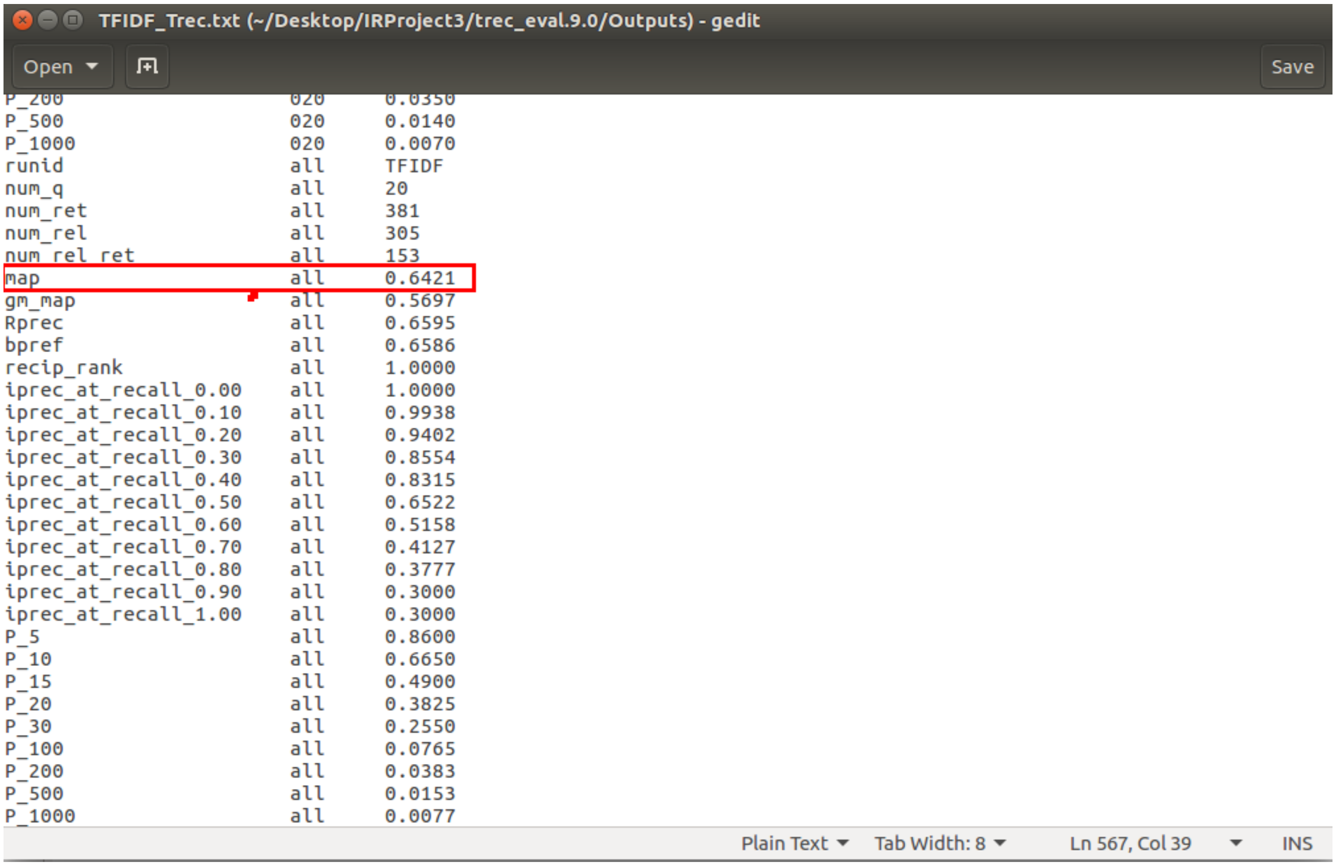


TFIDF_Results.txt (~/Desktop/IRProject3/trec_eval.9.0/Outputs) - gedit
Open
Save

```
001 Q1 653941482882134016 1 5.2120166 TFIDF
001 Q1 653278466788487168 2 3.0028787 TFIDF
001 Q1 653278493485236224 3 2.6648602 TFIDF
001 Q1 653941285045276672 4 2.402888 TFIDF
001 Q1 653278355677184000 5 2.402303 TFIDF
001 Q1 653278331278913536 6 2.402303 TFIDF
001 Q1 654279454278197248 7 2.1423473 TFIDF
001 Q1 653278536707506176 8 2.133669 TFIDF
001 Q1 653941326854160385 9 2.1318882 TFIDF
001 Q1 654279231854243840 10 2.1318882 TFIDF
001 Q1 653941528214306816 11 1.8740418 TFIDF
001 Q1 654279376385765376 12 1.6053727 TFIDF
001 Q1 654266183605063680 13 1.3217103 TFIDF
001 Q1 654279404542140416 14 1.2842982 TFIDF
001 Q1 654279248019103744 15 1.2842982 TFIDF
001 Q1 654279238258958336 16 1.2842982 TFIDF
001 Q1 653278473516138496 17 1.1295207 TFIDF
001 Q1 653278409238319104 18 1.1237609 TFIDF
001 Q1 654279374187982848 19 0.6251782 TFIDF
001 Q1 653929195400261632 20 0.59237486 TFIDF
002 Q2 654279412926517249 1 11.416957 TFIDF
002 Q2 653941469523214336 2 1.7607087 TFIDF
002 Q2 653941400904536064 3 1.6269076 TFIDF
```

MAP Value via TREC_eval: We run trec_eval on our query results via command line using the below command.

```
./trec_eval -q -c -M3440 Outputs/qrel.txt Outputs/TFIDF_Results.txt > Outputs/TFIDF_Trec.txt
```

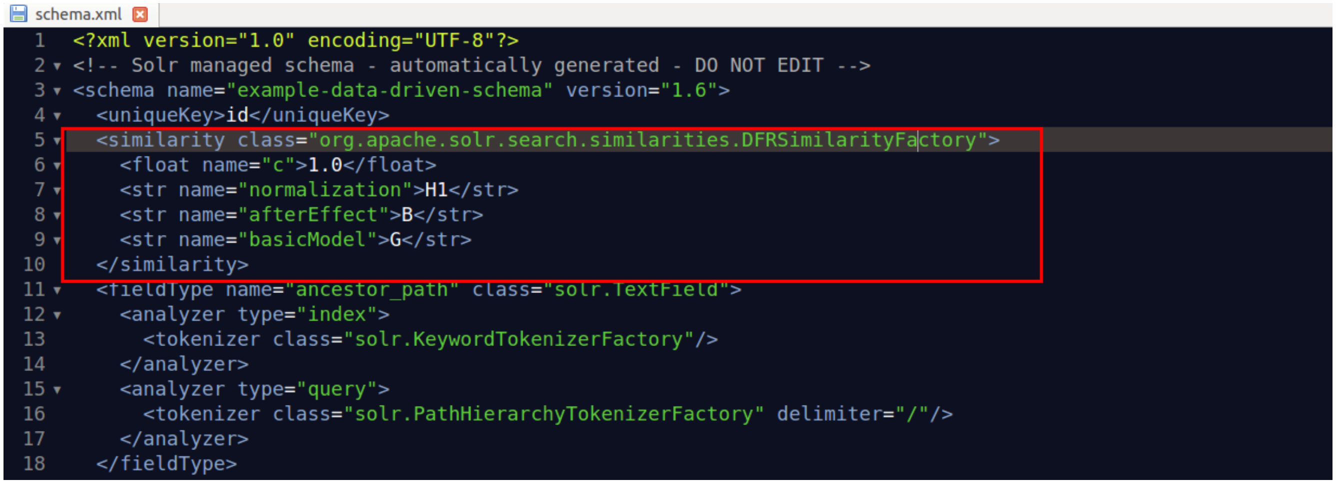

TFIDF_Trec.txt (~/Desktop/IRProject3/trec_eval.9.0/Outputs) - gedit
Open
Save
Plain Text
Tab Width: 8
Ln 567, Col 39
INS

```
P_200                   020     0.0350
P_500                   020     0.0140
P_1000                  020     0.0070
runid                   all     TFIDF
num_q                   all     20
num_ret                 all     381
num_rel                 all     305
num_rel_ret             all     153
map                     all     0.6421
gm_map                  all     0.5697
Rprec                   all     0.6595
bpref                   all     0.6586
recip_rank              all     1.0000
iprec_at_recall_0.00    all     1.0000
iprec_at_recall_0.10    all     0.9938
iprec_at_recall_0.20    all     0.9402
iprec_at_recall_0.30    all     0.8554
iprec_at_recall_0.40    all     0.8315
iprec_at_recall_0.50    all     0.6522
iprec_at_recall_0.60    all     0.5158
iprec_at_recall_0.70    all     0.4127
iprec_at_recall_0.80    all     0.3777
iprec_at_recall_0.90    all     0.3000
iprec_at_recall_1.00    all     0.3000
P_5                     all     0.8600
P_10                    all     0.6650
P_15                    all     0.4900
P_20                    all     0.3825
P_30                    all     0.2550
P_100                   all     0.0765
P_200                   all     0.0383
P_500                   all     0.0153
P_1000                  all     0.0077
```

ii. Divergence Form Randomness

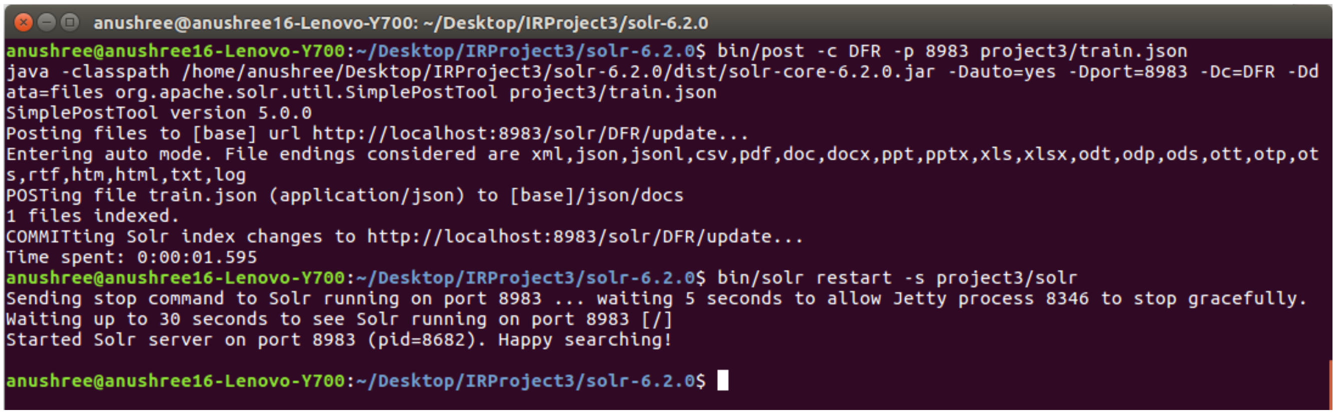
URL:https://lucene.apache.org/core/5_5_0/core/org/apache/lucene/search/similarities/DFRSimilarity.html

Modifications in Schema.xml: For the DFR model, we have taken "BasicModelG" plus "Bernoulli" first normalization plus "H2" second normalization as per instructions in Project pdf.



```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Solr managed schema - automatically generated - DO NOT EDIT -->
<schema name="example-data-driven-schema" version="1.6">
  <uniqueKey>id</uniqueKey>
  <similarity class="org.apache.solr.search.similarities.DFRSimilarityFactory">
    <float name="c">1.0</float>
    <str name="normalization">H1</str>
    <str name="afterEffect">B</str>
    <str name="basicModel">G</str>
  </similarity>
  <fieldType name="ancestor_path" class="solr.TextField">
    <analyzer type="index">
      <tokenizer class="solr.KeywordTokenizerFactory"/>
    </analyzer>
    <analyzer type="query">
      <tokenizer class="solr.PathHierarchyTokenizerFactory" delimiter="/"/>
    </analyzer>
  </fieldType>
```

Indexing data to Solr Query Results: We index the train.json with the newly changed schema.xml in Solr using terminal commands.

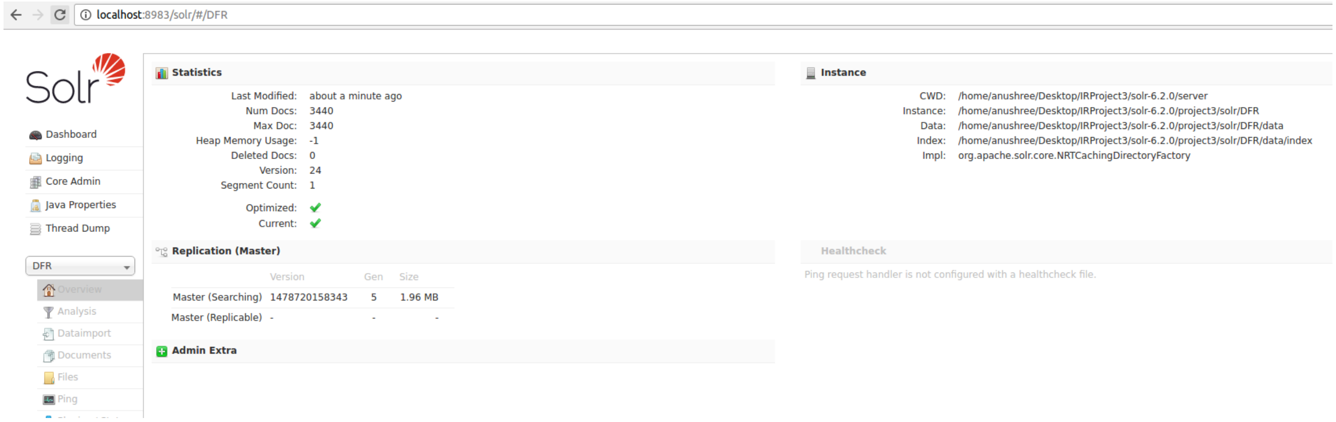


anushree@anushree16-Lenovo-Y700: ~/Desktop/IRProject3/solr-6.2.0

```
anushree@anushree16-Lenovo-Y700:~/Desktop/IRProject3/solr-6.2.0$ bin/post -c DFR -p 8983 project3/train.json
java -classpath /home/anushree/Desktop/IRProject3/solr-6.2.0/dist/solr-core-6.2.0.jar -Dauto=yes -Dport=8983 -Dc=DFR -Dd
ata=files org.apache.solr.util.SimplePostTool project3/train.json
SimplePostTool version 5.0.0
Posting files to [base] url http://localhost:8983/solr/DFR/update...
Entering auto mode. File endings considered are xml,json,jsonl,csv,pdf,doc,docx,ppt,pptx,xls,xlsx,odt,odp,ods,ott,otp,ot
s,rtf,htm,html,txt,log
POSTing file train.json (application/json) to [base]/json/docs
1 files indexed.
COMMITting Solr index changes to http://localhost:8983/solr/DFR/update...
Time spent: 0:00:01.595
anushree@anushree16-Lenovo-Y700:~/Desktop/IRProject3/solr-6.2.0$ bin/solr restart -s project3/solr
Sending stop command to Solr running on port 8983 ... waiting 5 seconds to allow Jetty process 8346 to stop gracefully.
Waiting up to 30 seconds to see Solr running on port 8983 [/]
Started Solr server on port 8983 (pid=8682). Happy searching!

anushree@anushree16-Lenovo-Y700:~/Desktop/IRProject3/solr-6.2.0$ 
```

Verifying if the documents are indexed on frontend.



Statistics

Last Modified: about a minute ago
Num Docs: 3440
Max Doc: 3440
Heap Memory Usage: -1
Deleted Docs: 0
Version: 24
Segment Count: 1
Optimized: ✓
Current: ✓

Replication (Master)

	Version	Gen	Size
Master (Searching)	1478720158343	5	1.96 MB
Master (Replicable)	-	-	-

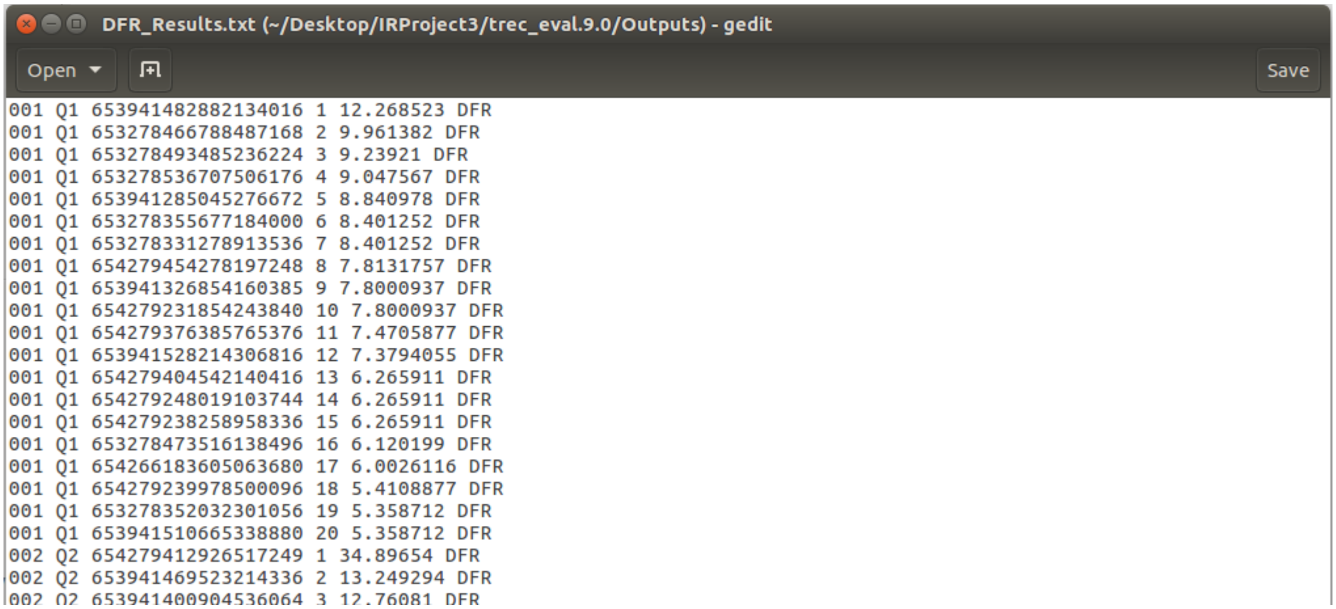
Instance

CWD: /home/anushree/Desktop/IRProject3/solr-6.2.0/server
Instance: /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/DFR
Data: /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/DFR/data
Index: /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/DFR/data/index
Impl: org.apache.solr.core.NRTCachingDirectoryFactory

Healthcheck

Ping request handler is not configured with a healthcheck file.

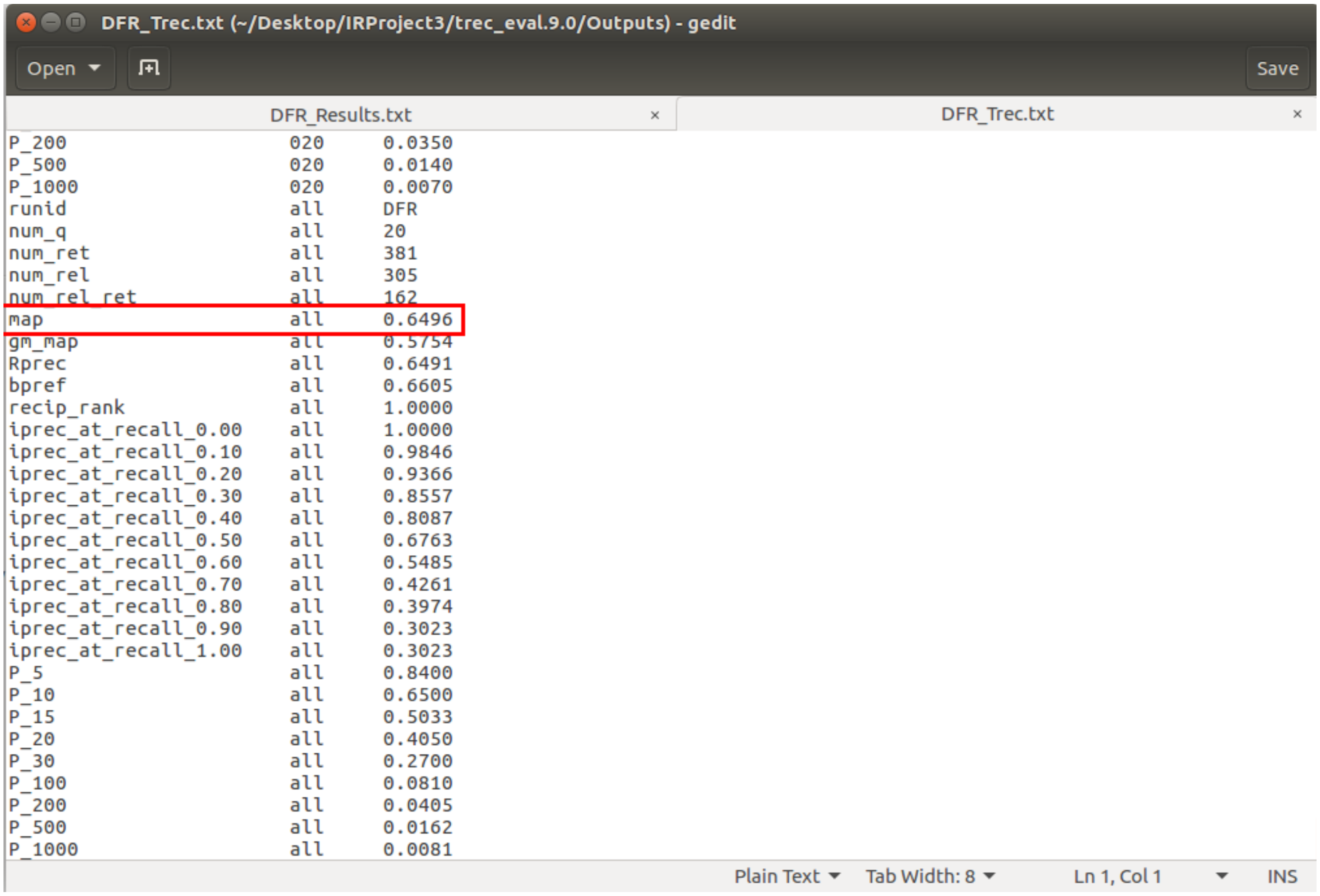
Query Results: Next we run the json_to_trec.py to get the output in format of trec input.


DFR_Results.txt (~/Desktop/IRProject3/trec_eval.9.0/Outputs) - gedit
Open
Save

```
001 Q1 653941482882134016 1 12.268523 DFR
001 Q1 653278466788487168 2 9.961382 DFR
001 Q1 653278493485236224 3 9.23921 DFR
001 Q1 653278536707506176 4 9.047567 DFR
001 Q1 653941285045276672 5 8.840978 DFR
001 Q1 653278355677184000 6 8.401252 DFR
001 Q1 653278331278913536 7 8.401252 DFR
001 Q1 654279454278197248 8 7.8131757 DFR
001 Q1 653941326854160385 9 7.8000937 DFR
001 Q1 654279231854243840 10 7.8000937 DFR
001 Q1 654279376385765376 11 7.4705877 DFR
001 Q1 653941528214306816 12 7.3794055 DFR
001 Q1 654279404542140416 13 6.265911 DFR
001 Q1 654279248019103744 14 6.265911 DFR
001 Q1 654279238258958336 15 6.265911 DFR
001 Q1 653278473516138496 16 6.120199 DFR
001 Q1 654266183605063680 17 6.0026116 DFR
001 Q1 654279239978500096 18 5.4108877 DFR
001 Q1 653278352032301056 19 5.358712 DFR
001 Q1 653941510665338880 20 5.358712 DFR
002 Q2 654279412926517249 1 34.89654 DFR
002 Q2 653941469523214336 2 13.249294 DFR
002 Q2 653941400904536064 3 12.76081 DFR
```

MAP Value via TREC_eval: We run trec_eval on our query results via command line using the below command.

```
./trec_eval -q -c -M3440 Outputs/qrel.txt Outputs/DFR_Results.txt > Outputs/DFR_Trec.txt
```

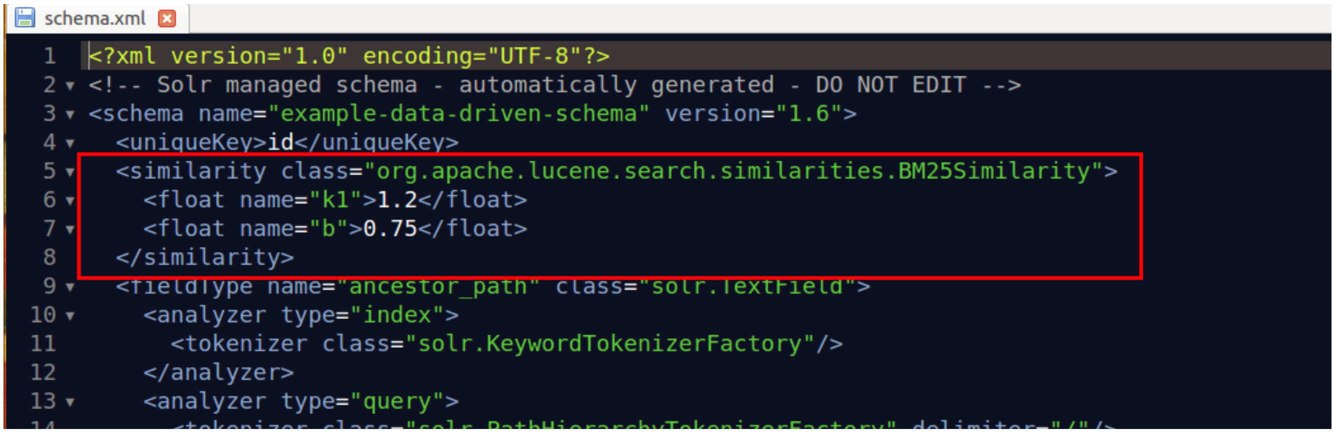


```
P_200                   020     0.0350
P_500                   020     0.0140
P_1000                  020     0.0070
runid                   all     DFR
num_q                   all     20
num_ret                 all     381
num_rel                 all     305
num_rel_ret             all     162
map                     all     0.6496
gm_map                  all     0.5754
Rprec                   all     0.6491
bpref                   all     0.6605
recip_rank              all     1.0000
iprec_at_recall_0.00    all     1.0000
iprec_at_recall_0.10    all     0.9846
iprec_at_recall_0.20    all     0.9366
iprec_at_recall_0.30    all     0.8557
iprec_at_recall_0.40    all     0.8087
iprec_at_recall_0.50    all     0.6763
iprec_at_recall_0.60    all     0.5485
iprec_at_recall_0.70    all     0.4261
iprec_at_recall_0.80    all     0.3974
iprec_at_recall_0.90    all     0.3023
iprec_at_recall_1.00    all     0.3023
P_5                     all     0.8400
P_10                    all     0.6500
P_15                    all     0.5033
P_20                    all     0.4050
P_30                    all     0.2700
P_100                   all     0.0810
P_200                   all     0.0405
P_500                   all     0.0162
P_1000                  all     0.0081
```

iii. **BM25**

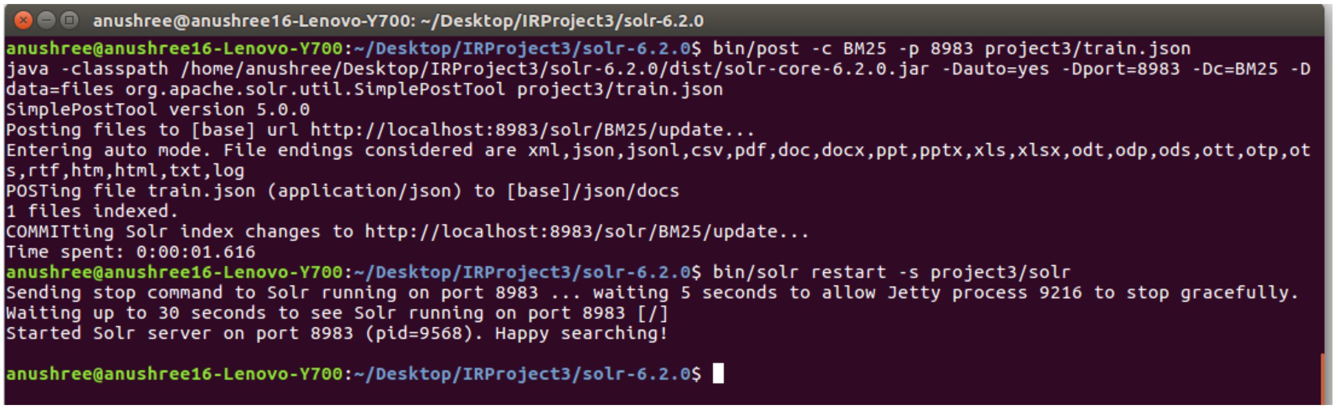
URL:https://lucene.apache.org/core/5_5_0/core/org/apache/lucene/search/similarities/DFRSimilarity.html

Modifications in Schema.xml: For the BM25 Model, we have implemented as below with default values for k1 as 1.2 and b as 0.75.



```xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- Solr managed schema - automatically generated - DO NOT EDIT -->
<schema name="example-data-driven-schema" version="1.6">
  <uniqueKey>id</uniqueKey>
  <similarity class="org.apache.lucene.search.similarities.BM25Similarity">
    <float name="k1">1.2</float>
    <float name="b">0.75</float>
  </similarity>
  <fieldType name="ancestor_path" class="solr.TextField">
    <analyzer type="index">
      <tokenizer class="solr.KeywordTokenizerFactory"/>
    </analyzer>
    <analyzer type="query">
      <tokenizer class="solr.PathHierarchyTokenizerFactory" delimiter="/"/>
```

Indexing data to Solr Query Results: We index the train.json with the newly changed schema.xml in Solr using terminal commands.

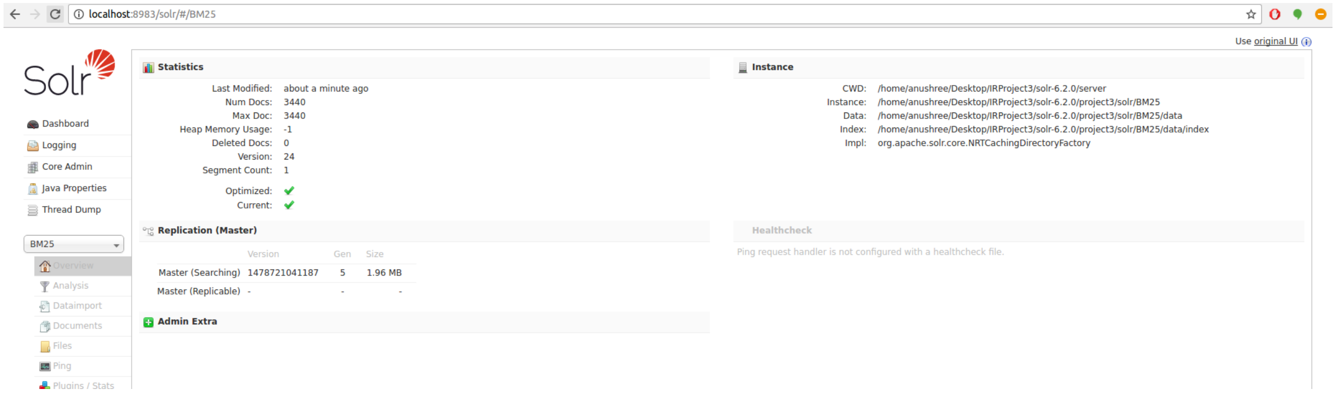


anushree@anushree16-Lenovo-Y700: ~/Desktop/IRProject3/solr-6.2.0

```
anushree@anushree16-Lenovo-Y700:~/Desktop/IRProject3/solr-6.2.0$ bin/post -c BM25 -p 8983 project3/train.json
java -classpath /home/anushree/Desktop/IRProject3/solr-6.2.0/dist/solr-core-6.2.0.jar -Dauto=yes -Dport=8983 -Dc=BM25 -D
data=files org.apache.solr.util.SimplePostTool project3/train.json
SimplePostTool version 5.0.0
Posting files to [base] url http://localhost:8983/solr/BM25/update...
Entering auto mode. File endings considered are xml,json,jsonl,csv,pdf,doc,docx,ppt,pptx,xls,xlsx,odt,odp,ods,ott,otp,ot
s,rtf,htm,html,txt,log
POSTing file train.json (application/json) to [base]/json/docs
1 files indexed.
COMMITting Solr index changes to http://localhost:8983/solr/BM25/update...
Time spent: 0:00:01.616
anushree@anushree16-Lenovo-Y700:~/Desktop/IRProject3/solr-6.2.0$ bin/solr restart -s project3/solr
Sending stop command to Solr running on port 8983 ... waiting 5 seconds to allow Jetty process 9216 to stop gracefully.
Waiting up to 30 seconds to see Solr running on port 8983 [/]
Started Solr server on port 8983 (pid=9568). Happy searching!

anushree@anushree16-Lenovo-Y700:~/Desktop/IRProject3/solr-6.2.0$
```

Verifying if the documents are indexed on frontend.



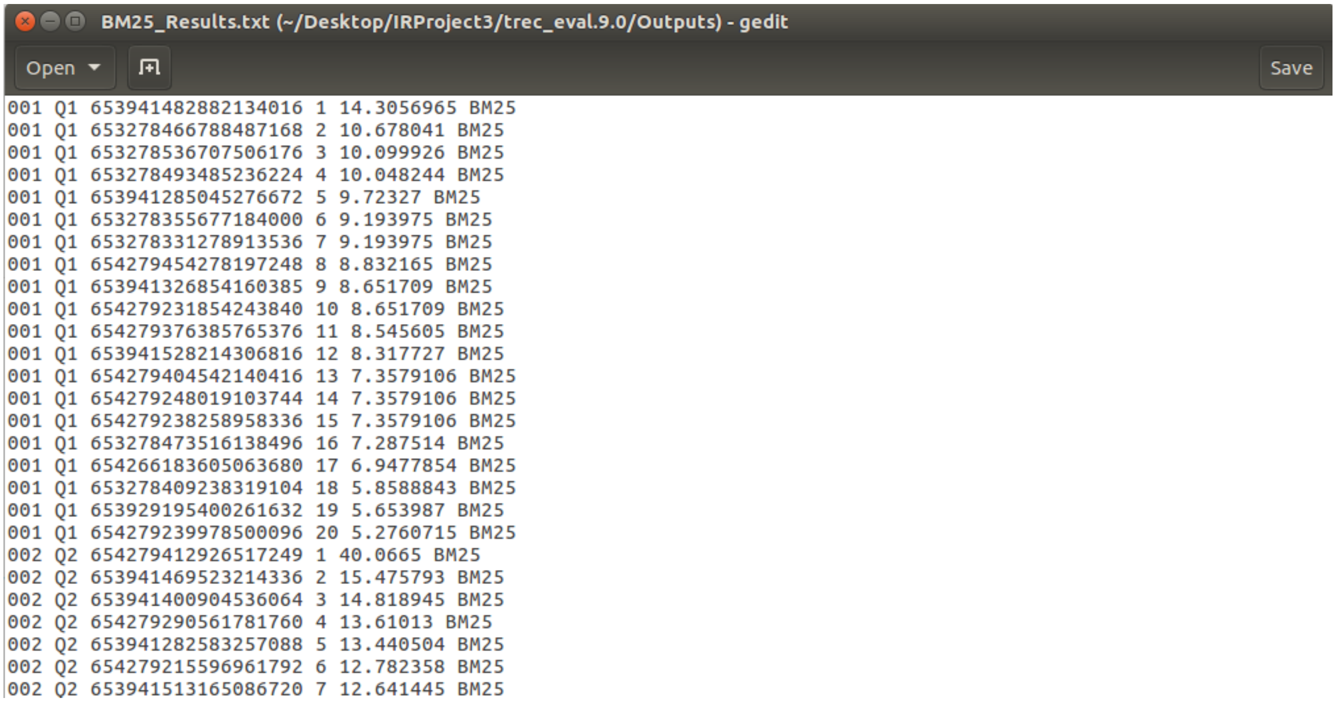
Last Modified: about a minute ago
Num Docs: 3440
Max Doc: 3440
Heap Memory Usage: -1
Deleted Docs: 0
Version: 24
Segment Count: 1
Optimized: ✓
Current: ✓

CWD: /home/anushree/Desktop/IRProject3/solr-6.2.0/server
Instance: /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/BM25
Data: /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/BM25/data
Index: /home/anushree/Desktop/IRProject3/solr-6.2.0/project3/solr/BM25/data/index
Impl: org.apache.solr.core.NRTCachingDirectoryFactory

	Version	Gen	Size
Master (Searching)	1478721041187	5	1.96 MB
Master (Replicable)	-	-	-

Ping request handler is not configured with a healthcheck file.

Query Results: Next we run the json_to_trec.py to get the output in format of trec input.

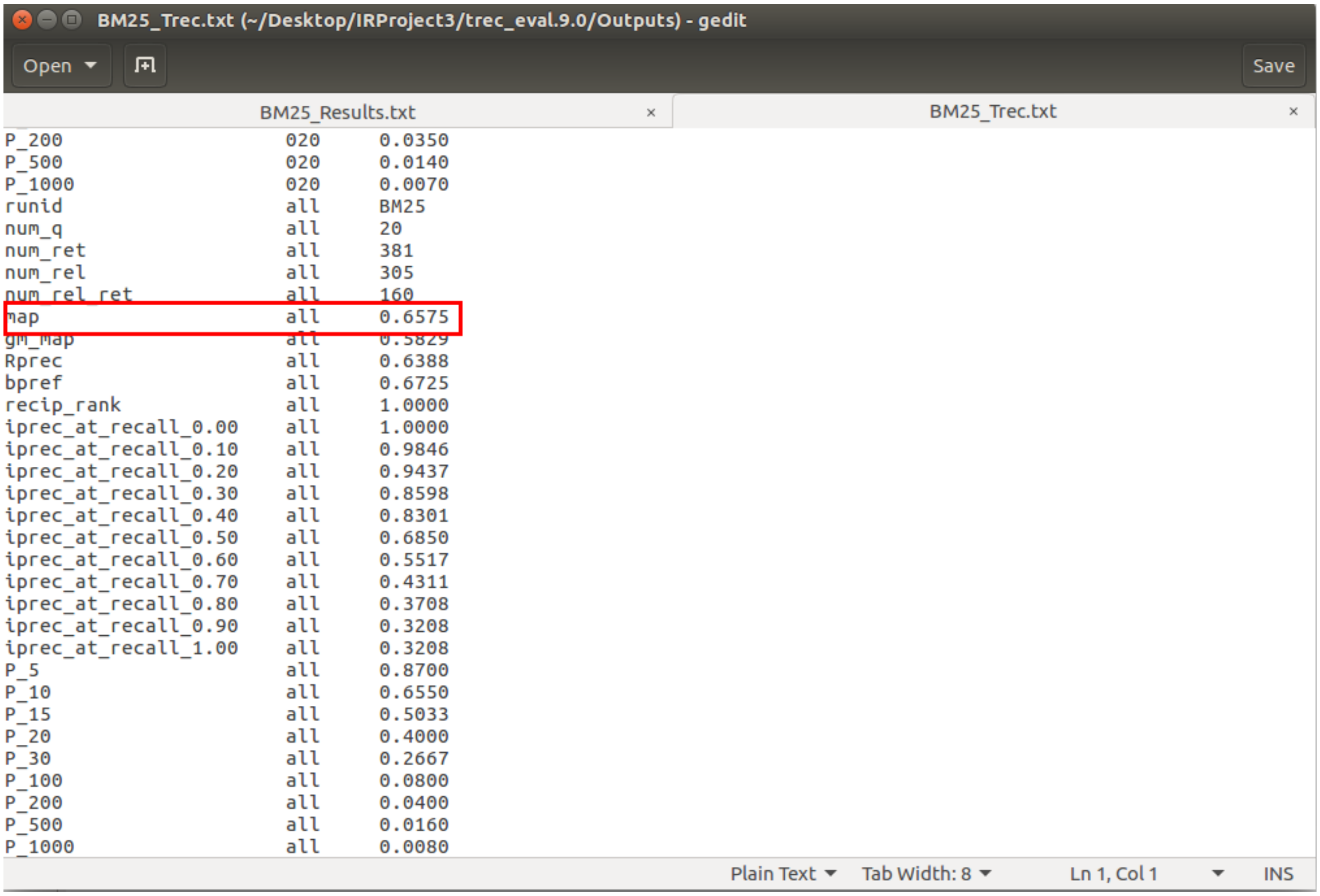


BM25_Results.txt (~/Desktop/IRProject3/trec_eval.9.0/Outputs) - gedit
Open
Save

```
001 Q1 653941482882134016 1 14.3056965 BM25
001 Q1 653278466788487168 2 10.678041 BM25
001 Q1 653278536707506176 3 10.099926 BM25
001 Q1 653278493485236224 4 10.048244 BM25
001 Q1 653941285045276672 5 9.72327 BM25
001 Q1 653278355677184000 6 9.193975 BM25
001 Q1 653278331278913536 7 9.193975 BM25
001 Q1 654279454278197248 8 8.832165 BM25
001 Q1 653941326854160385 9 8.651709 BM25
001 Q1 654279231854243840 10 8.651709 BM25
001 Q1 654279376385765376 11 8.545605 BM25
001 Q1 653941528214306816 12 8.317727 BM25
001 Q1 654279404542140416 13 7.3579106 BM25
001 Q1 654279248019103744 14 7.3579106 BM25
001 Q1 654279238258958336 15 7.3579106 BM25
001 Q1 653278473516138496 16 7.287514 BM25
001 Q1 654266183605063680 17 6.9477854 BM25
001 Q1 653278409238319104 18 5.8588843 BM25
001 Q1 653929195400261632 19 5.653987 BM25
001 Q1 654279239978500096 20 5.2760715 BM25
002 Q2 654279412926517249 1 40.0665 BM25
002 Q2 653941469523214336 2 15.475793 BM25
002 Q2 653941400904536064 3 14.818945 BM25
002 Q2 654279290561781760 4 13.61013 BM25
002 Q2 653941282583257088 5 13.440504 BM25
002 Q2 654279215596961792 6 12.782358 BM25
002 Q2 653941513165086720 7 12.641445 BM25
```

MAP Value via TREC_eval: We run trec_eval on our query results via command line using the below command.

```
./trec_eval -q -c -M3440 Outputs/qrel.txt Outputs/BM25_Results.txt > Outputs/BM25_Trec.txt
```



```
P_200                   020     0.0350
P_500                   020     0.0140
P_1000                  020     0.0070
runid                   all     BM25
num_q                   all     20
num_ret                 all     381
num_rel                 all     305
num_rel_ret             all     160
map                     all     0.6575
gm_map                  all     0.5829
Rprec                   all     0.6388
bpref                   all     0.6725
recip_rank              all     1.0000
iprec_at_recall_0.00    all     1.0000
iprec_at_recall_0.10    all     0.9846
iprec_at_recall_0.20    all     0.9437
iprec_at_recall_0.30    all     0.8598
iprec_at_recall_0.40    all     0.8301
iprec_at_recall_0.50    all     0.6850
iprec_at_recall_0.60    all     0.5517
iprec_at_recall_0.70    all     0.4311
iprec_at_recall_0.80    all     0.3708
iprec_at_recall_0.90    all     0.3208
iprec_at_recall_1.00    all     0.3208
P_5                     all     0.8700
P_10                    all     0.6550
P_15                    all     0.5033
P_20                    all     0.4000
P_30                    all     0.2667
P_100                   all     0.0800
P_200                   all     0.0400
P_500                   all     0.0160
P_1000                  all     0.0080
```

Summary of MAP Values for all the Models:

Model Name	MAP Values via TREC_Eval
VSM – Classic Similarity	0.6418
VSM – Sweet Spot Similarity Factory	0.6421
DFR Similarity	0.6496
BM25 Similarity	0.6575

PART 2: What have you done to improve the performance in terms of MAP (and maybe also other measures)? Please list what you have done one by one and present why you do this, what the effect is before and after your intervention. You are suggested to use tables or plots to make the comparison informative and clear.

We have made various modifications in all the three models and noted the changes in MAP values with respect to the default MAP values noted in PART 1 of this project.

1. **Parameter Tuning with Dismax Query Parser**

Idea: DFR and BM25 models have few parameters which can be modified to have an impact on the MAP value for a given collection of data.

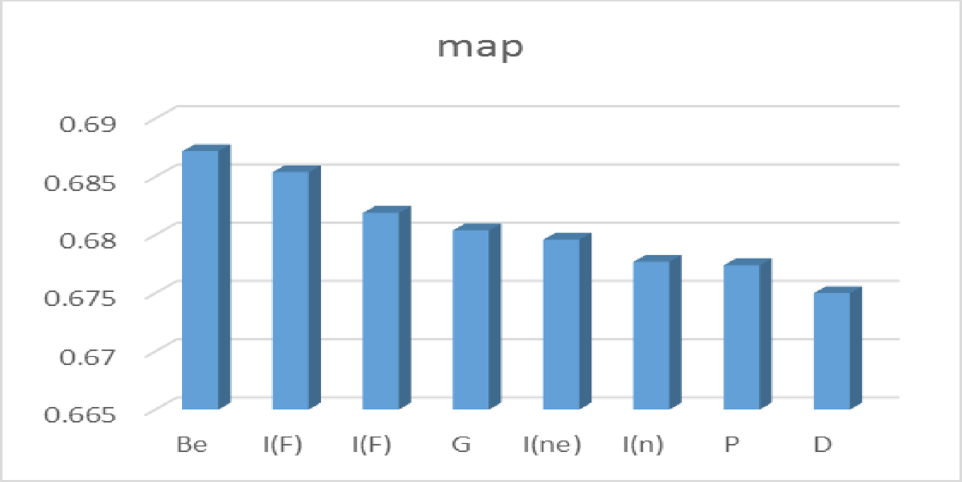
Implementation:
We tested all the models for different values of all parameters by modifying the json_to_trec.py code to automatically update these parameters in schema.xml for certain ranges and and reindexes data with new changes in Solr. We also impemented Dismax Parser with the queries. Below are the parameters for the models which we have tunned to get the optimal values for the given data.

Observations:

1) DFR Similarity:
The optimized value of MAP we got by tuning parameters is **0.6872** which is for the below parameter values:

- Basic Model: Be
- AfterEffect: B
- Normalization: H2

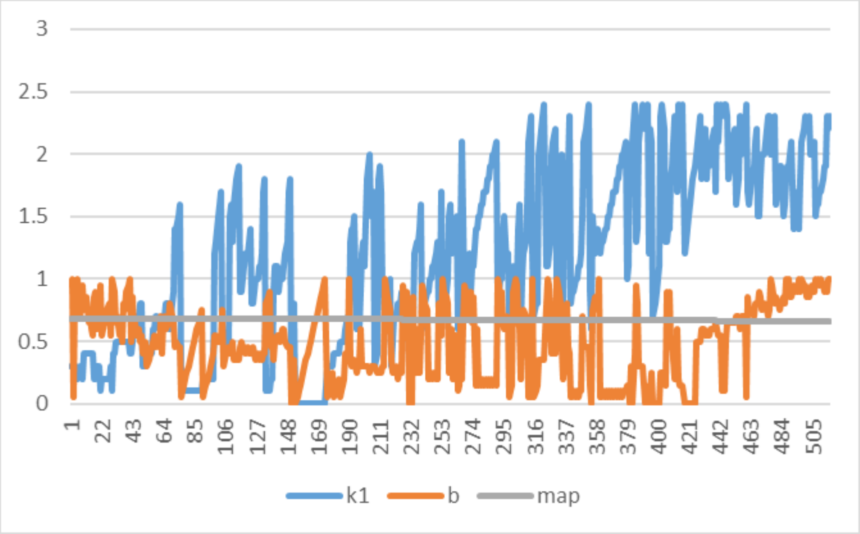
Below is plot for DFR with afterEffect = B and normalization = H2 kept constant.



2) BM25 Similarity:
This similarity has two free parameters - k1 and b. The optimized value of MAP we got by tuning the above parameter is **0.6853** for the below parameter values:

- k1: 0.3
- b: 1/ 0.95

Below is the plot for BM25 plotted against various values of k1 and b parameters.



2. Query Expansion via Language Translation

Idea: Translating the input query to all the three given languages i.e., English, German and Russian will increase the relevant documents returned even if in some other languages.

Implementation: We modified the given json_to_trec.py code to use Google Translator API to translate the English query to Russian and German and vice versa. The problem we faced here was that the Google API allows only one translation when we access it via code. Thus finally we manually translated all the queries and passed them to json_to_trec.py code for all the three models. Below are the screenshots and summary of the MAP scores we obtained for all the three models with respect to default MAP values.

Observations:

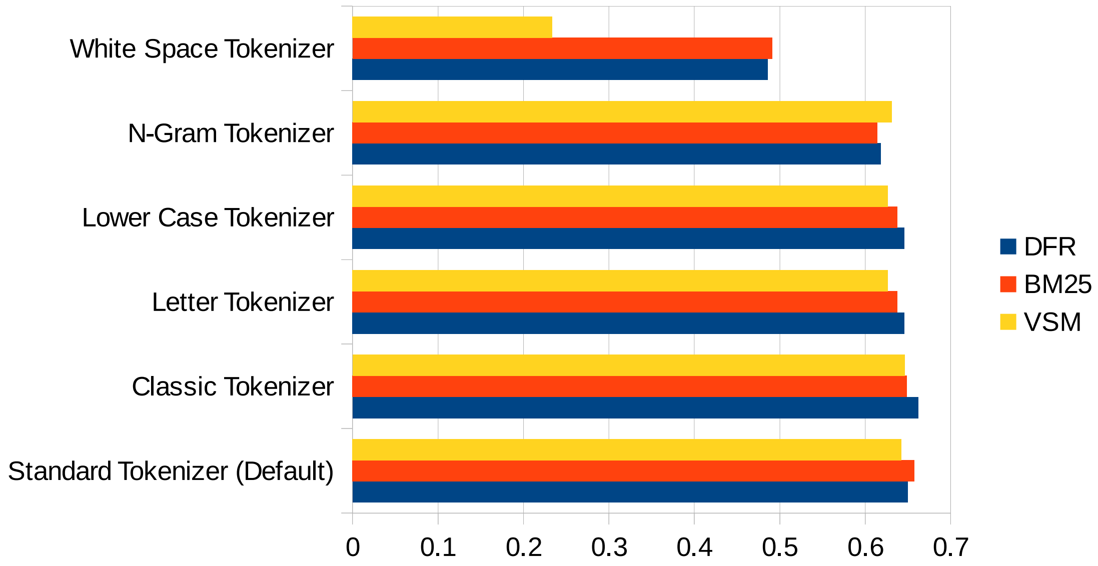
Model Name	MAP Values via TREC_Eval - Default	MAP Values via TREC_Eval - Modified	Change
VSM – Classic Similarity	0.6418	0.6450	+0.0032
VSM – Sweet Spot Similarity	0.6421	0.6474	+0.0053
DFR Similarity	0.6496	0.6557	+0.0061
BM25 Similarity	0.6575	0.6502	-0.0073

Thus by implementing the language translation of queries we increased the MAP value for VSM and DFR models.

3. Tokenization

Idea: Tokenizers are responsible for breaking the input text into tokens. Thus the way we form tokens from documents while indexing and from the query while searching can have an impact on the number of relevant documents returned.

Implementation: We implemented all the Tokenizers with all the three models as shown in the charts below. Charts are plotted for MAP Values vs Tokenizers.



Thus by implementing tokenizers we can conclude that Classic Tokenizer works best for DFR and Standard Tokenizer works best for VSM and BM25.

4. **Query Expansion using Synonyms with Dismax Parser**

Idea: While indexing as well as query parsing if we use synonyms for the tokens there is a possibility of returning more relevant documents.

Implementation: We updated the synonym.txt which is placed in solr\confg of the core with all the relevant synonyms based on the indexed documents and the queries provided. Since we got good MAP values for all models using Dismax Parser we implemented that as well.

Model Name	MAP Values via TREC_Eval - Default	MAP Values via TREC_Eval - Modified	Change
VSM – Sweet Spot Similarity	0.6421	0.6784	+0.0363
DFR Similarity	0.6496	0.6754	+0.0258
BM25 Similarity	0.6575	0.6783	+0.0208

Using synonyms alongwith Dismax parser for token boost the relevant documents returned to a great extent for all the three models. Max change we could observe for Vector Space Model.