Ch. Sravan Kumar
UBIT Name: Scheekat

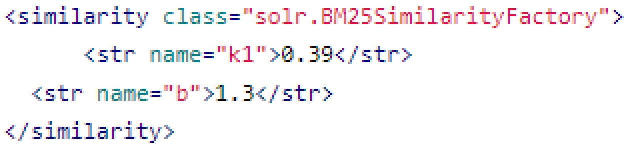
CSE 535 - INFORMATION RETRIEVAL PROJECT-3

Evaluation of IR Models

Implementing the Default Configurations of the IR Models

BM25 Model:

Using the following Similarity class in the schema.xml file we can implement BM25 model

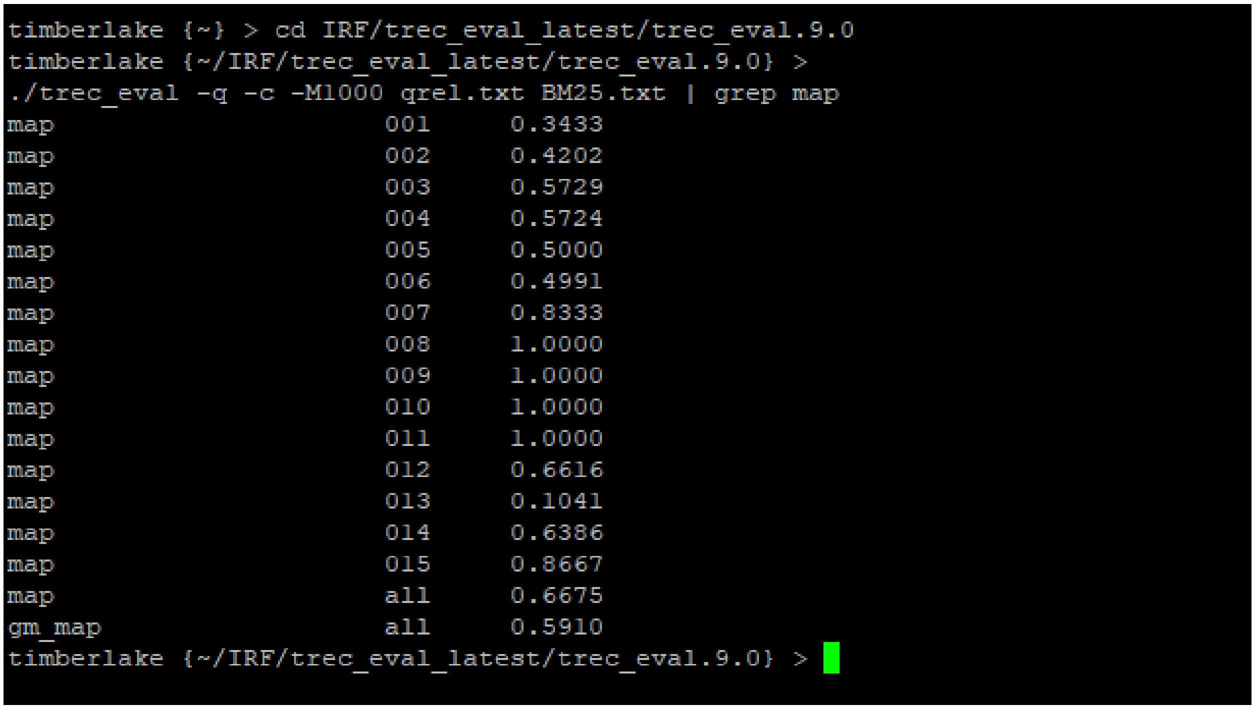


```
<similarity class="solr.BM25SimilarityFactory">
      <str name="k1">0.39</str>
  <str name="b">1.3</str>
</similarity>
```

After re-indexing with train.json provided for the above configured schema.xml file for the Core on solr, we run the TREC_eval to get the MAP value for the test queries provided to us along with their manual judgement file qrel.txt

We use the below to get the BM25 MAP value for the default configuration:

./trec_eval -q -c -M1000 qrel.txt BM25.txt | grep map



```
timberlake {~} > cd IRF/trec_eval_latest/trec_eval.9.0
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
./trec_eval -q -c -M1000 qrel.txt BM25.txt | grep map
map                     001     0.3433
map                     002     0.4202
map                     003     0.5729
map                     004     0.5724
map                     005     0.5000
map                     006     0.4991
map                     007     0.8333
map                     008     1.0000
map                     009     1.0000
map                     010     1.0000
map                     011     1.0000
map                     012     0.6616
map                     013     0.1041
map                     014     0.6386
map                     015     0.8667
map                     all     0.6675
gm_map                  all     0.5910
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

Ch. Sravan Kumar
UBIT Name: Scheekat

VSM (Vector space model):

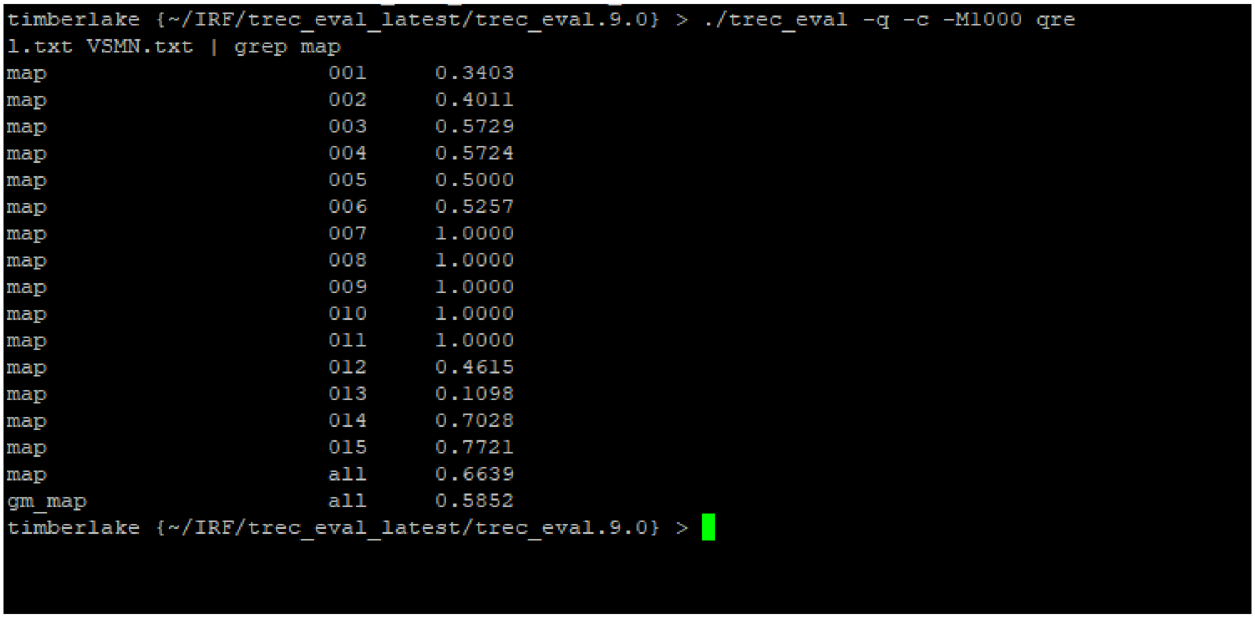
Using the following Similarity class in the schema.xml file we can implement the Vector Space Model as a global configuration:

```
<similarity class="solr.ClassicSimilarityFactory"/>
```

After re-indexing with train.json provided for the above configured schema.xml file for the Core on solr, we run the TREC_eval to get the MAP value for the test queries provided to us along with their manual judgement file qrel.txt

We use the below to get the VSM MAP value for the default configuration:

```
./trec_eval -q -c -M1000 qrel.txt VSNM.txt | grep map
```



```
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} > ./trec_eval -q -c -M1000 qre
l.txt VSMN.txt | grep map
map                     001     0.3403
map                     002     0.4011
map                     003     0.5729
map                     004     0.5724
map                     005     0.5000
map                     006     0.5257
map                     007     1.0000
map                     008     1.0000
map                     009     1.0000
map                     010     1.0000
map                     011     1.0000
map                     012     0.4615
map                     013     0.1098
map                     014     0.7028
map                     015     0.7721
map                     all     0.6639
gm_map                  all     0.5852
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

DFR(Divergence from Randomness):

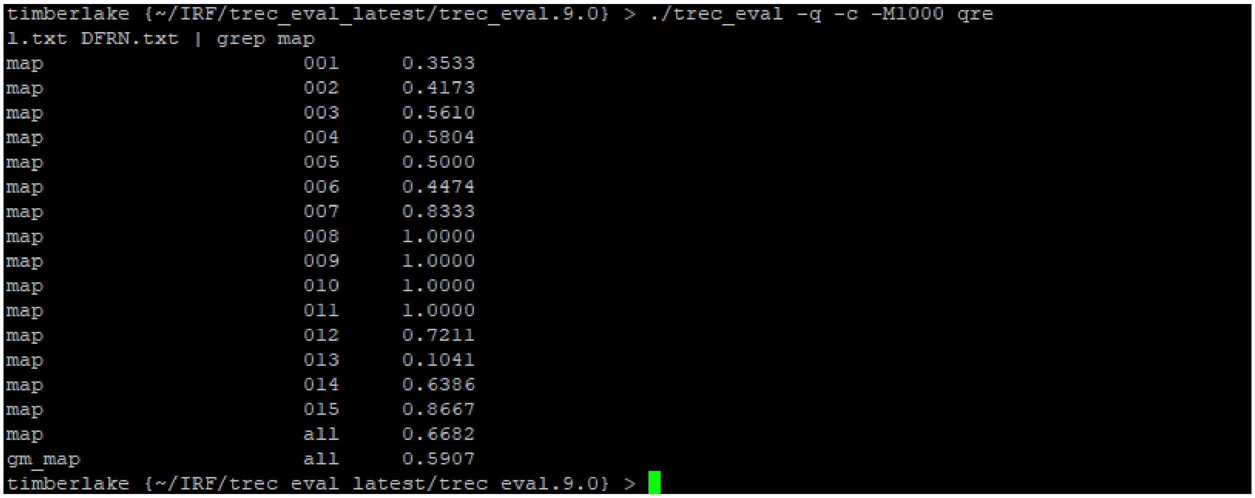
Using the following Similarity class in the schema.xml file we can implement the divergence from randomness as a global configuration:

```
<similarity class="solr.DFRSimilarityFactory">
      <str name="afterEffect">B</str>
      <str name="basicModel">G</str>
      <str name="normalization">H2</str>
</similarity>
```

After re-indexing with train.json provided for the above configured schema.xml file for the Core on solr, we run the TREC_eval to get the MAP value for the test queries provided to us along with their manual judgement file qrel.txt

We use the below to get the DFR MAP value for the default configuration:

./trec_eval -q -c -M1000 qrel.txt DFRN.txt | grep map

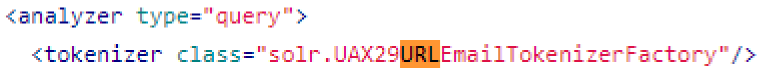


```
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} > ./trec_eval -q -c -M1000 qre
l.txt DFRN.txt | grep map
map                     001     0.3533
map                     002     0.4173
map                     003     0.5610
map                     004     0.5804
map                     005     0.5000
map                     006     0.4474
map                     007     0.8333
map                     008     1.0000
map                     009     1.0000
map                     010     1.0000
map                     011     1.0000
map                     012     0.7211
map                     013     0.1041
map                     014     0.6386
map                     015     0.8667
map                     all     0.6682
gm_map                  all     0.5907
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

Steps taken to Improve the performance:

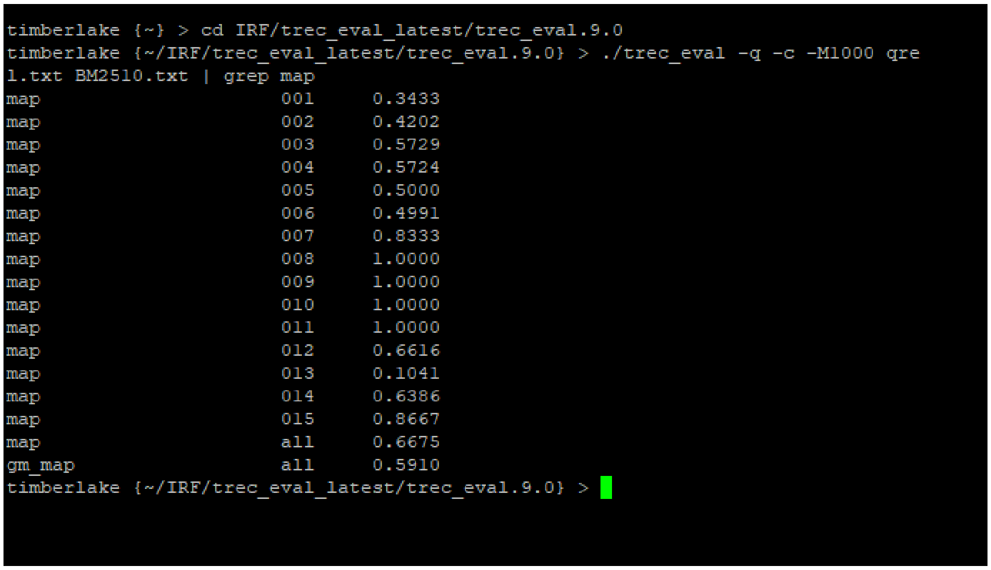
BM25:

1) **Experimented by adding URLTokenizer instead of standard tokenizer for text_en for analyzer type Query:**



```
<analyzer type="query">
  <tokenizer class="solr.UAX29URLEmailTokenizerFactory"/>
```

After re-indexing the value still remains same as above. No change in the map value.



```
timberlake {~} > cd IRF/trec_eval_latest/trec_eval.9.0
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} > ./trec_eval -q -c -M1000 qre
l.txt BM2510.txt | grep map
map                     001     0.3433
map                     002     0.4202
map                     003     0.5729
map                     004     0.5724
map                     005     0.5000
map                     006     0.4991
map                     007     0.8333
map                     008     1.0000
map                     009     1.0000
map                     010     1.0000
map                     011     1.0000
map                     012     0.6616
map                     013     0.1041
map                     014     0.6386
map                     015     0.8667
map                     all     0.6675
gm_map                  all     0.5910
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

2) Experimented by removing with charFilter to remove “#” and “@” in both the index and query for text_en Query:

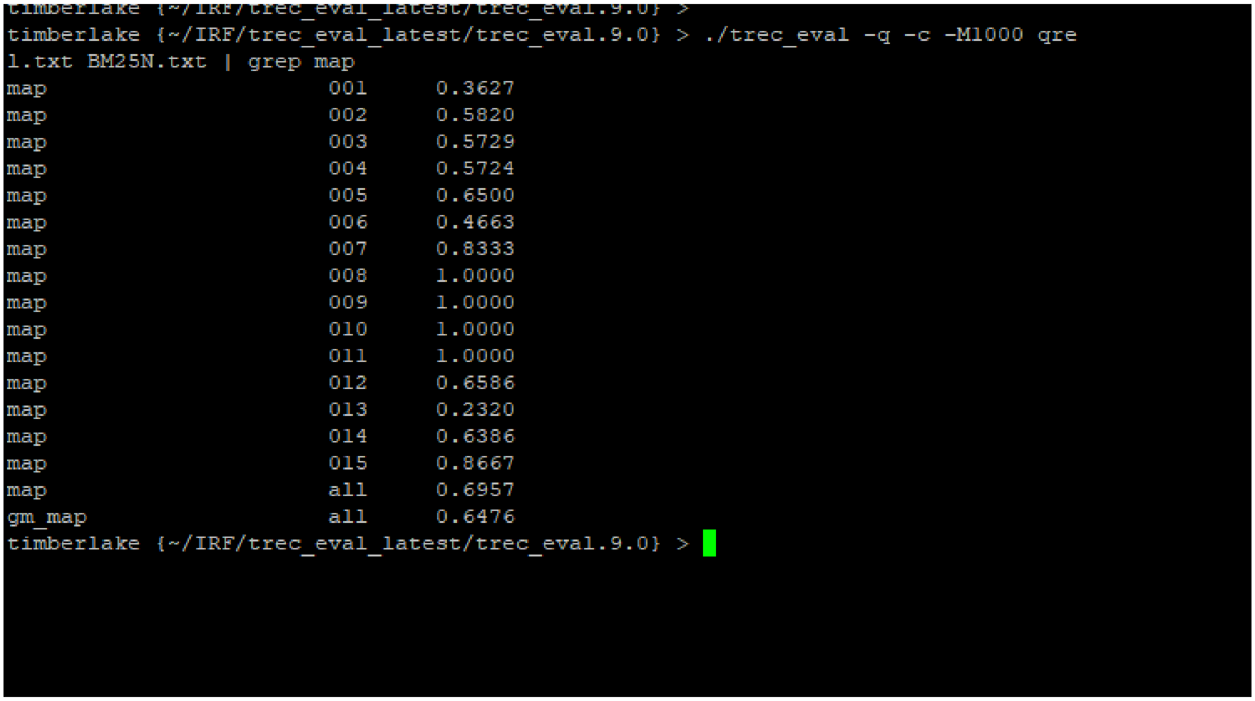
```
 <analyzer type="index">
 <charFilterclass="solr.PatternReplaceCharFilterFactory"pattern="([@#])"replacement=""/>
 </analyzer type>

<analyzer type="query">
<charFilterclass="solr.PatternReplaceCharFilterFactory"pattern="([@#])"replacement=">
 </analyzer type>
```

MAP values is more when no filter was used, but then the values decreased when the filter was used. Map value decreased from 0.6675 to 0.6528. So filters are removed from the schema.

3)Using query expansion and translating queries

By using query expansion and synonyms match for the given test data, Noticed that the MAP values increased.

Map Value increased from 0.6675 to 0.6957.

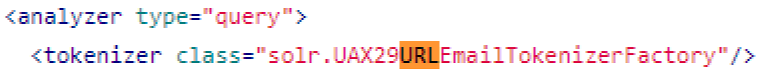


```
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} > ./trec_eval -q -c -M1000 qre
l.txt BM25N.txt | grep map
map                     001     0.3627
map                     002     0.5820
map                     003     0.5729
map                     004     0.5724
map                     005     0.6500
map                     006     0.4663
map                     007     0.8333
map                     008     1.0000
map                     009     1.0000
map                     010     1.0000
map                     011     1.0000
map                     012     0.6586
map                     013     0.2320
map                     014     0.6386
map                     015     0.8667
map                     all     0.6957
gm_map                  all     0.6476
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

Ch. Sravan Kumar
UBIT Name: Scheekat

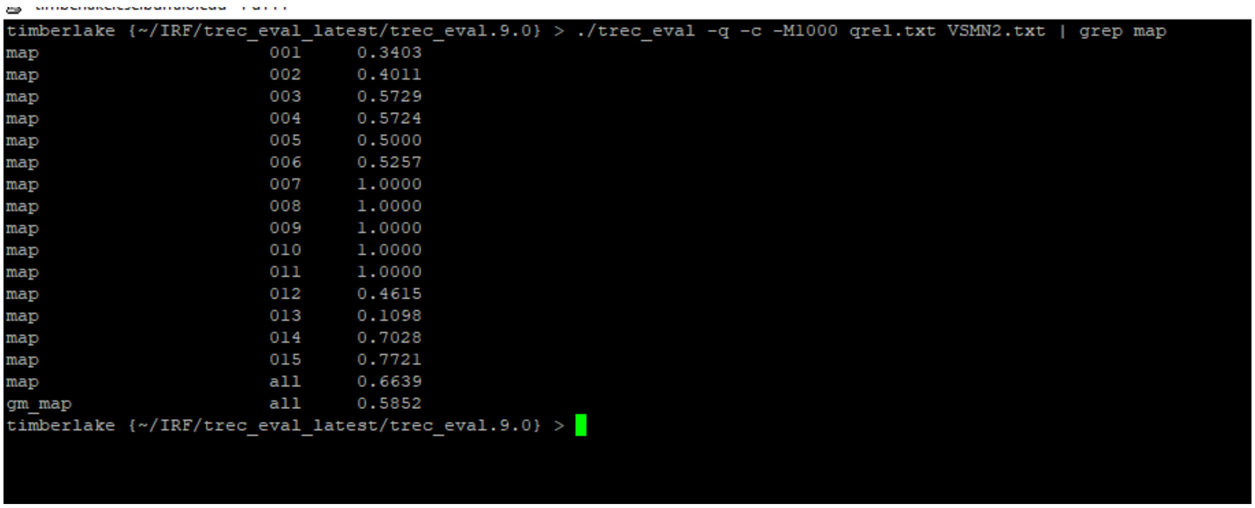
VSM (Vector space model):

1) Experimented by adding URLTokenizer instead of standard tokenizer for text_en for analyzer type Query:



```
<analyzer type="query">
  <tokenizer class="solr.UAX29URLEmailTokenizerFactory"/>
```

After re-indexing the value still remains same as above. No change in the map value.

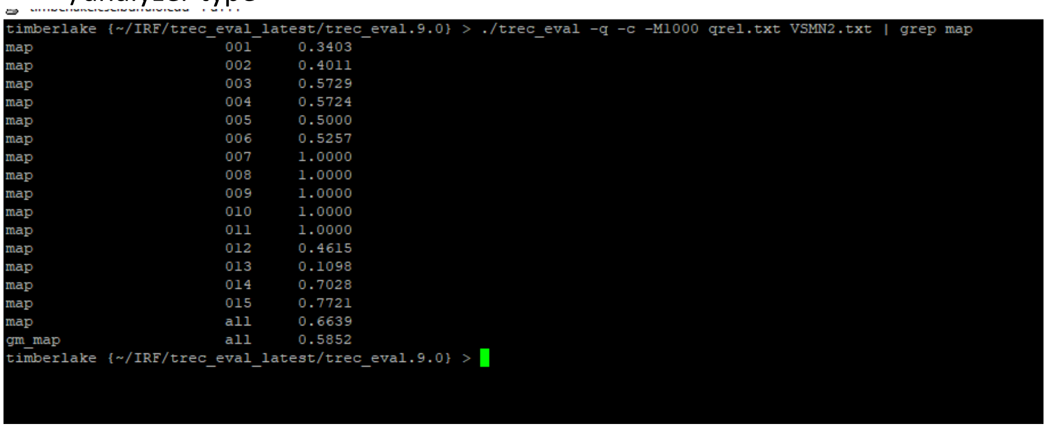


```
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} > ./trec_eval -q -c -M1000 qrel.txt VSMN2.txt | grep map
map                     001     0.3403
map                     002     0.4011
map                     003     0.5729
map                     004     0.5724
map                     005     0.5000
map                     006     0.5257
map                     007     1.0000
map                     008     1.0000
map                     009     1.0000
map                     010     1.0000
map                     011     1.0000
map                     012     0.4615
map                     013     0.1098
map                     014     0.7028
map                     015     0.7721
map                     all     0.6639
gm_map                  all     0.5852
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

2)Experimented by removing with charFilter to remove “#” and “@” in both the index and query for text_en Query:

<analyzer type="index">
<charFilter class="solr.PatternReplaceCharFilterFactory"pattern="([@#])" replacement=""/>
</analyzer type>

<analyzer type="query">
<charFilter class="solr.PatternReplaceCharFilterFactory"pattern="([@#])" replacement=""/>
</analyzer type>

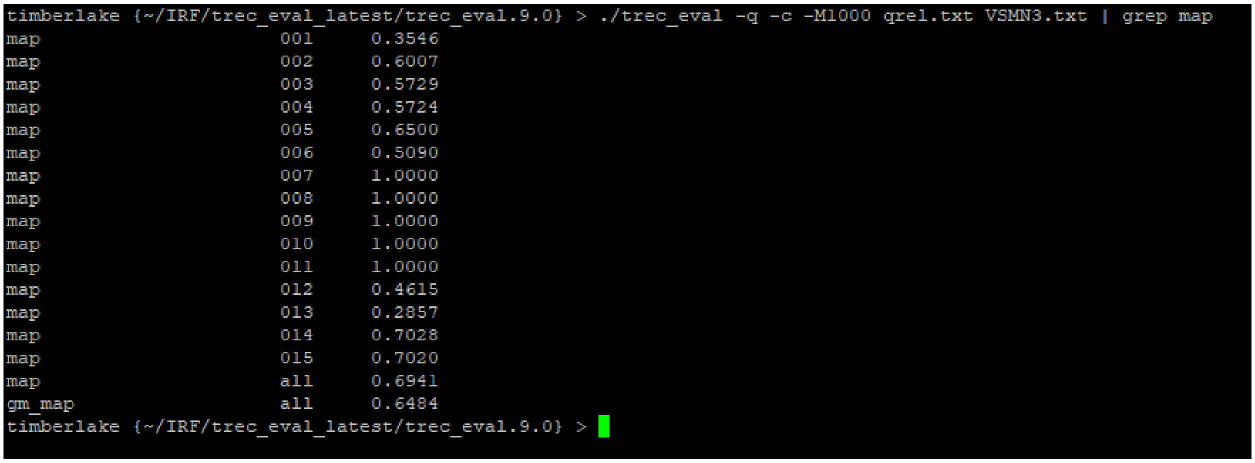


```
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} > ./trec_eval -q -c -M1000 qrel.txt VSMN2.txt | grep map
map                     001     0.3403
map                     002     0.4011
map                     003     0.5729
map                     004     0.5724
map                     005     0.5000
map                     006     0.5257
map                     007     1.0000
map                     008     1.0000
map                     009     1.0000
map                     010     1.0000
map                     011     1.0000
map                     012     0.4615
map                     013     0.1098
map                     014     0.7028
map                     015     0.7721
map                     all     0.6639
gm_map                  all     0.5852
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

By using both the techniques the value remains same in VSM model.

3)Using query expansion and translating queries

By using query expansion and synonyms match for the given test data, Noticed that the MAP values increased.
By this method MAP value increased from 0.6639 to 0.6941.

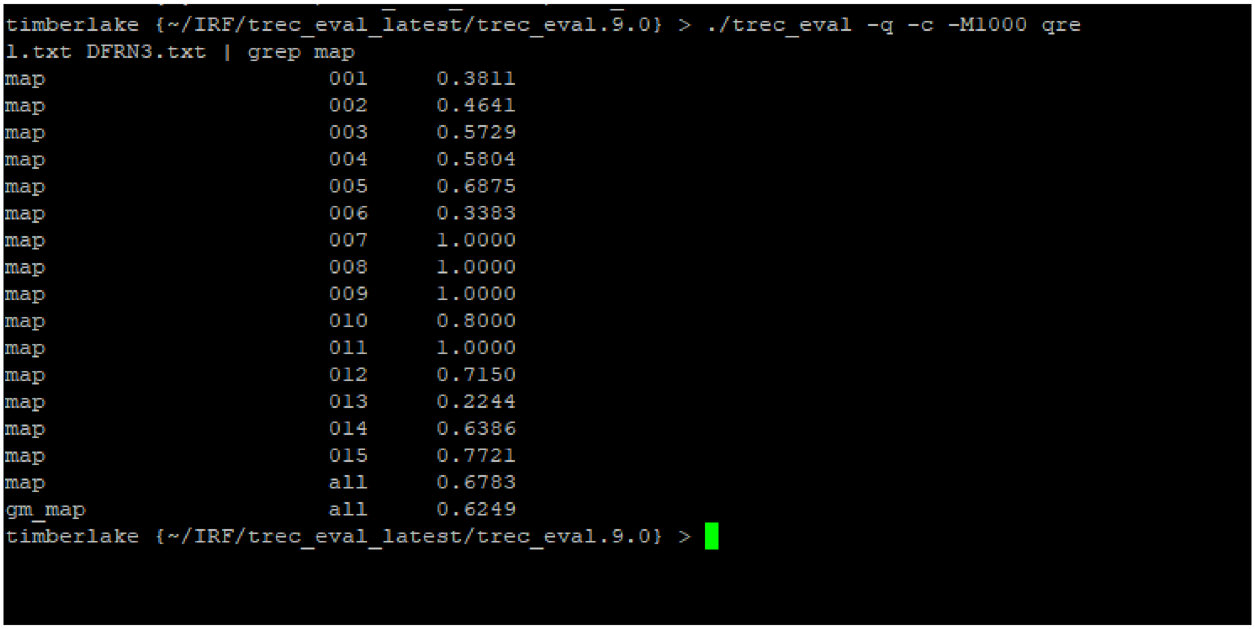


```
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} > ./trec_eval -q -c -M1000 qrel.txt VSMN3.txt | grep map
map                     001     0.3546
map                     002     0.6007
map                     003     0.5729
map                     004     0.5724
map                     005     0.6500
map                     006     0.5090
map                     007     1.0000
map                     008     1.0000
map                     009     1.0000
map                     010     1.0000
map                     011     1.0000
map                     012     0.4615
map                     013     0.2857
map                     014     0.7028
map                     015     0.7020
map                     all     0.6941
gm_map                  all     0.6484
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

DFR(Divergence from Random) model:

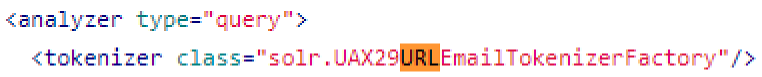
1) **Using query expansion and translating queries**

   By using query expansion and synonyms match for the given test data, Noticed that the MAP values increased.
   Observed that MAP value is increased from 0.6682 to 0.6783



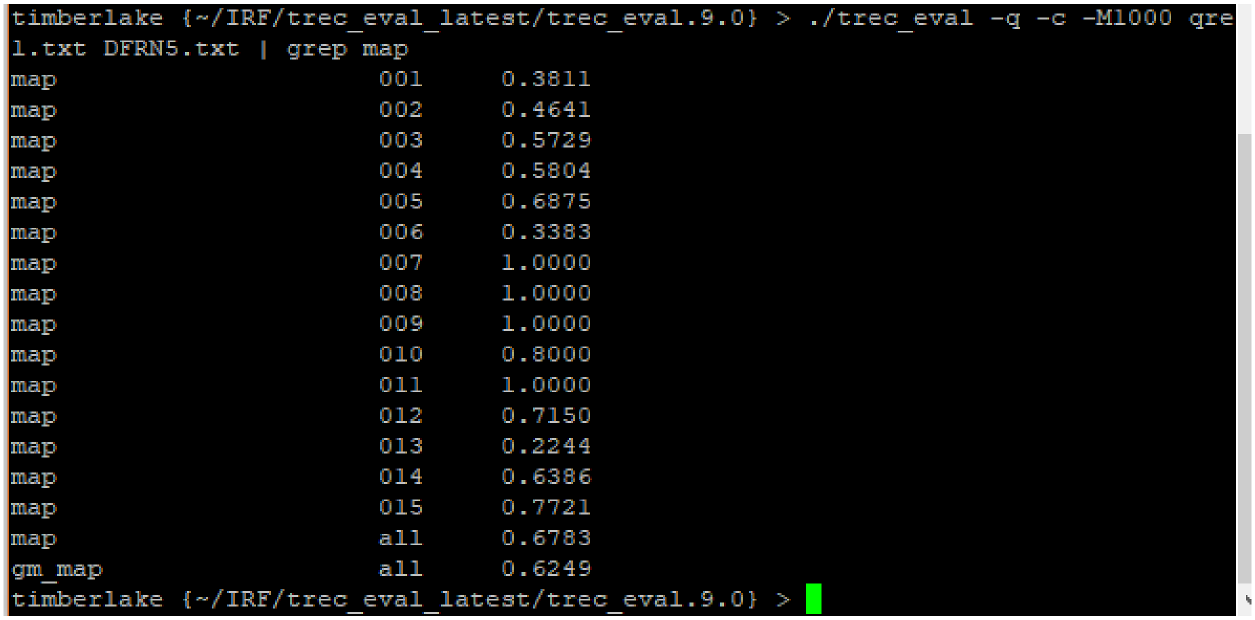
```
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} > ./trec_eval -q -c -M1000 qre
l.txt DFRN3.txt | grep map
map                     001     0.3811
map                     002     0.4641
map                     003     0.5729
map                     004     0.5804
map                     005     0.6875
map                     006     0.3383
map                     007     1.0000
map                     008     1.0000
map                     009     1.0000
map                     010     0.8000
map                     011     1.0000
map                     012     0.7150
map                     013     0.2244
map                     014     0.6386
map                     015     0.7721
map                     all     0.6783
gm_map                  all     0.6249
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

1) Experimented by adding URLTokenizer instead of standard tokenizer for text_en for analyzer type Query:



```
<analyzer type="query">
  <tokenizer class="solr.UAX29URLEmailTokenizerFactory"/>
```

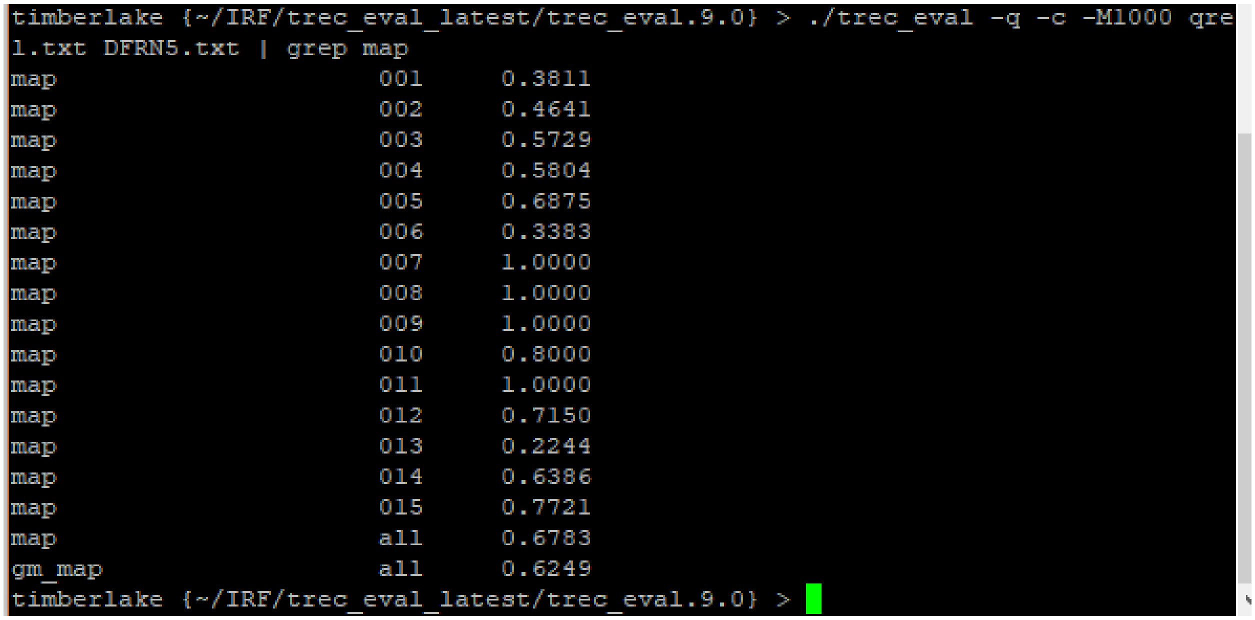
After re-indexing the value still remains same as above. No change in the map value.



```
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} > ./trec_eval -q -c -M1000 qre
l.txt DFRN5.txt | grep map
map                     001     0.3811
map                     002     0.4641
map                     003     0.5729
map                     004     0.5804
map                     005     0.6875
map                     006     0.3383
map                     007     1.0000
map                     008     1.0000
map                     009     1.0000
map                     010     0.8000
map                     011     1.0000
map                     012     0.7150
map                     013     0.2244
map                     014     0.6386
map                     015     0.7721
map                     all     0.6783
gm_map                  all     0.6249
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

2)Experimented by removing with charFilter to remove “#” and “@” in both the index and query for text_en Query:

```
<analyzer type="index">
<charFilter class="solr.PatternReplaceCharFilterFactory"pattern="([@#])" replacement=""/>
</analyzer type>

<analyzer type="query">
<charFilter class="solr.PatternReplaceCharFilterFactory"pattern="([@#])" replacement=""/>
</analyzer type>
```



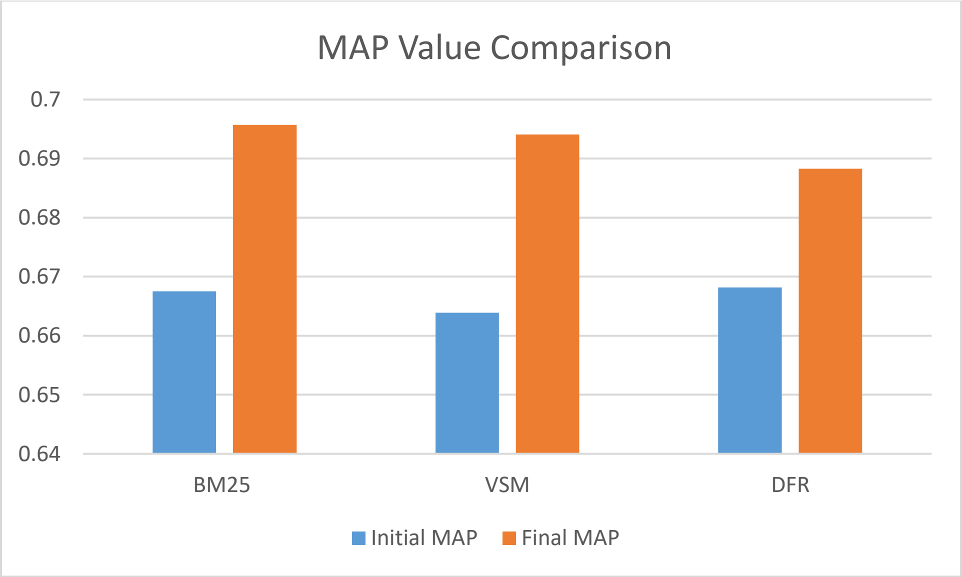
```
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} > ./trec_eval -q -c -M1000 qre
l.txt DFRN5.txt | grep map
map                     001     0.3811
map                     002     0.4641
map                     003     0.5729
map                     004     0.5804
map                     005     0.6875
map                     006     0.3383
map                     007     1.0000
map                     008     1.0000
map                     009     1.0000
map                     010     0.8000
map                     011     1.0000
map                     012     0.7150
map                     013     0.2244
map                     014     0.6386
map                     015     0.7721
map                     all     0.6783
gm_map                  all     0.6249
timberlake {~/IRF/trec_eval_latest/trec_eval.9.0} >
```

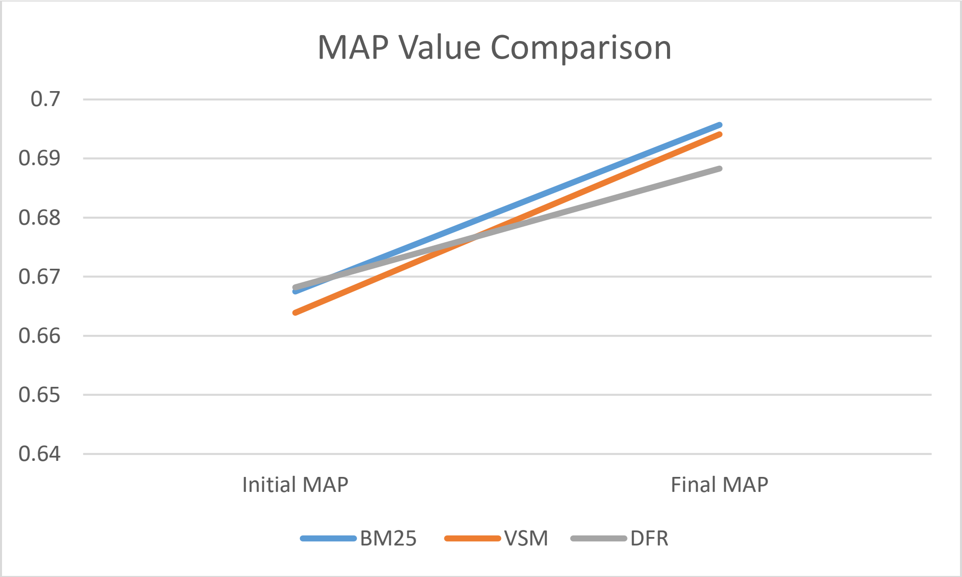
By using both the techniques the value remains same in DFR model.

Result Summary:

Therefore after the optimization of the default model settings we obtained the following as MAP values:

IR MODEL	Initial MAP Value	Final MAP value
Okapi BM25	0.6675	0.6957
Vector Space Model (VSM)	0.6639	0.6941
Divergence from Random(DFR)	0.6682	0.6883





Name: Ch. Sravan Kumar
UBIT Name: Scheekat