PROJECT 3

NAME	**:**	**MANASA RAO CHAKUNTA**
COURSE	**:**	**CSE-535 FALL 2020(INFORMATION RETRIEVAL)**
UB #	**:**	**50338196**
UBIT NAME	**:**	**manasara**

EVALUATION OF IR MODELS

ABSTRACT:

The goal of this project is to implement various IR models and evaluate their IR system based on Mean Average Precision(MAP) values. We are given twitter data in three languages - English, German and Russian, 15 sample queries and the corresponding relevance judgement. The given twitter data is being indexed on Solr using models:

- **BM25**
- **Vector Space Model**

and then evaluated by Trec_Eval program. The result obtained tells us how relevant our model was. The best MAP values obtained for models BM25 and VSM are 0.67 and -- respectively.

INTRODUCTION:

We have implemented the following IR Models: BM25 and Vector Space Model. We have used the given training queries provided to judge the Relevance with Precision and Recall of our results with the help of the TREC tool. Mean Average Precision (MAP) is used to judge how good the model is and we are taking those parameter values which are providing the highest map score in the model. Separate cores are created for each model to implement IR models. We have used schema.xml to define fields and field types and the similarity definition for each model.

MODEL IMPLEMENTATION STEPS:

- Firstly, create directories in FileZilla with core names (IRF20P3_BM25 and IRF20P3_VSM inside server.
- Create conf and data directories in these cores. Start Solr. A default core gettingstarted is created. Stop Solr.
- Copy the conf data from getting started to all the three cores. Start Solr.
- Do the indexing(bin/post) of train.json file on all the cores.
- Copy the managed_schema file to the local system and rename it to schema.xml. Stop Solr.
- Add similarity classes as per model to schema file. Copy changed schema file from local to the cores. Delete the previously created managed_schema file and schema.xml.bak before posting the data again.

- Start solr in Standalone. Post the train.json again using the new schema that has a similarity class as per the model. Now, run the json_to_trec.py script to get the output in trec_eval format. One text file is generated.
- Feed the generated output to trec_eval executable, to get MAP scores for the queries.
- Tweak the hyper-parameters in the similarity class to improve these MAP scores for each model.
- After selecting optimal values for hyperparameters, post the test-queries.txt file to generate the trec_eval formatted output which will be fed to the trec_eval executable later for relevance judgment.

MODELS:

BEST MATCHING 25(BM25): Okapi BM25 is a bag of words retrieval function which is used by search engines to rank the documents according to relevance of the documents to the queries. It was originally designed for short-length documents.

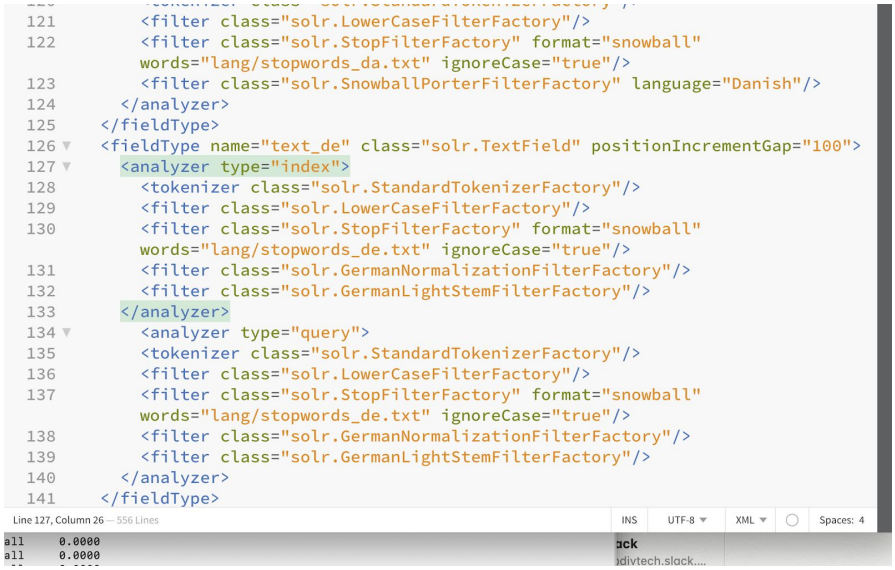
- For this model the default values of the hyper-parameters are k1 = 1.2 and b = 0.75
- The similarity class used is:

```
<similarity class="solr.BM25SimilarityFactory">

    <float name="k1">0.4</float>

    <float name="b">1.0</float>

 </similarity>
```

- Parameter k1 controls non-linear term frequency normalizations and b tells what degree documentation
- For larger text k1 tends to larger value, and for documents which touch on broader value b tends to be large.
- The following table shows various MAP values obtained from change of parameter k1 and b values.
- We have also used the dismax function to obtain more relevant documents.
- In the fieldtype for analyzser type=”index” and type=”query” are added which improved the MAP score. The default MAP score is **0.12.**

K1	b	MAP value
0.4	**1.0**	**0.7005**
1.0	1.0	0.6772
1.2	0.75	0.6756
0.5	0.9	0.6756

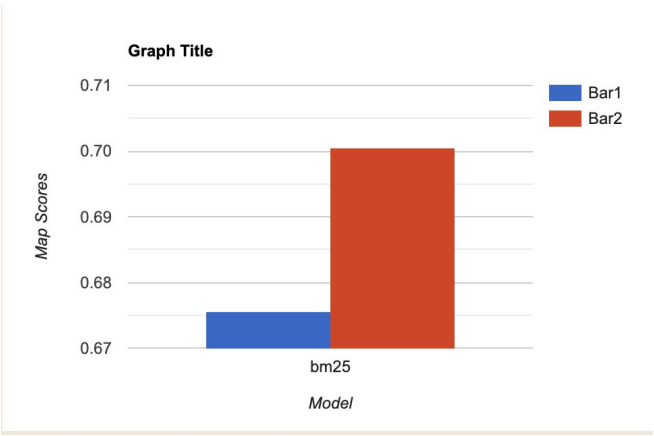
```
411         <tokenizer class="solr.StandardTokenizerFactory"/>
412         <filter class="solr.LowerCaseFilterFactory"/>
413         <filter class="solr.StopFilterFactory" words="lang/stopwords_ro.txt"
            ignoreCase="true"/>
414         <filter class="solr.SnowballPorterFilterFactory" language="Romanian"/>
415       </analyzer>
416     </fieldType>
417     <fieldType name="text_ru" class="solr.TextField" positionIncrementGap="100">
418       <analyzer type="index">
419         <tokenizer class="solr.StandardTokenizerFactory"/>
420         <filter class="solr.LowerCaseFilterFactory"/>
421         <filter class="solr.StopFilterFactory" format="snowball"
            words="lang/stopwords_ru.txt" ignoreCase="true"/>
422         <filter class="solr.SnowballPorterFilterFactory" language="Russian"/>
423       </analyzer>
424       <analyzer type="query">
425         <tokenizer class="solr.StandardTokenizerFactory"/>
426         <filter class="solr.LowerCaseFilterFactory"/>
427         <filter class="solr.StopFilterFactory" format="snowball"
            words="lang/stopwords_ru.txt" ignoreCase="true"/>
428         <filter class="solr.SnowballPorterFilterFactory" language="Russian"/>
429       </analyzer>
430     </fieldType>
```



```
121         <filter class="solr.LowerCaseFilterFactory"/>
122         <filter class="solr.StopFilterFactory" format="snowball"
            words="lang/stopwords_da.txt" ignoreCase="true"/>
123         <filter class="solr.SnowballPorterFilterFactory" language="Danish"/>
124       </analyzer>
125     </fieldType>
126     <fieldType name="text_de" class="solr.TextField" positionIncrementGap="100">
127       <analyzer type="index">
128         <tokenizer class="solr.StandardTokenizerFactory"/>
129         <filter class="solr.LowerCaseFilterFactory"/>
130         <filter class="solr.StopFilterFactory" format="snowball"
            words="lang/stopwords_de.txt" ignoreCase="true"/>
131         <filter class="solr.GermanNormalizationFilterFactory"/>
132         <filter class="solr.GermanLightStemFilterFactory"/>
133       </analyzer>
134       <analyzer type="query">
135         <tokenizer class="solr.StandardTokenizerFactory"/>
136         <filter class="solr.LowerCaseFilterFactory"/>
137         <filter class="solr.StopFilterFactory" format="snowball"
            words="lang/stopwords_de.txt" ignoreCase="true"/>
138         <filter class="solr.GermanNormalizationFilterFactory"/>
139         <filter class="solr.GermanLightStemFilterFactory"/>
140       </analyzer>
141     </fieldType>
Line 127, Column 26 — 556 Lines   INS   UTF-8   XML   Spaces: 4
all   0.0000
all   0.0000
```

```
<field name="lang" type="text_general"/>
<field name="text_de" type="text_de"/>
<field name="text_en" type="text_en"/>
<field name="text_ru" type="text_ru"/>
```

```
P_30                    015     0.4333
P_100                   015     0.1300
P_200                   015     0.0650
P_500                   015     0.0260
P_1000                  015     0.0130
runid                   all     bm25
num_q                   all     15
num_ret                 all     280
num_rel                 all     225
num_rel_ret             all     130
map                     all     0.7005
gm_map                  all     0.6329
Rprec                   all     0.6979
bpref                   all     0.7093
recip_rank              all     1.0000
iprec_at_recall_0.00    all     1.0000
iprec_at_recall_0.10    all     0.9667
iprec_at_recall_0.20    all     0.9286
iprec_at_recall_0.30    all     0.8875
iprec_at_recall_0.40    all     0.8595
iprec_at_recall_0.50    all     0.8233
iprec_at_recall_0.60    all     0.6745
iprec_at_recall_0.70    all     0.5418
iprec_at_recall_0.80    all     0.4457
iprec_at_recall_0.90    all     0.3353
iprec_at_recall_1.00    all     0.3353
P_5                     all     0.8533
P_10                    all     0.6600
P_15                    all     0.5156
P_20                    all     0.4333
P_30                    all     0.2889
P_100                   all     0.0867
P_200                   all     0.0433
P_500                   all     0.0173
P_1000                  all     0.0087
ubuntu@ip-172-31-21-12:~/solr-8.2.0/trec_eval-9.0.7$
```



VSM:

In the Vector Space model everything is going to be a vector in some high dimensional space(Words, documents and queries). Every word in the document is going to be a dimension.

- Highest MAP score obtained is **0.6756** after adding dismax function with fields pf and qf= text_en, text_de and text_ru
- The similarity class used in VSM is:

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

```
P_20                    015     0.6500
P_30                    015     0.4333
P_100                   015     0.1300
P_200                   015     0.0650
P_500                   015     0.0260
P_1000                  015     0.0130
runid                   all     vsm
num_q                   all     15
num_ret                 all     280
num_rel                 all     225
num_rel_ret             all     121
map                     all     0.6756
gm_map                  all     0.6082
Rprec                   all     0.6474
bpref                   all     0.6704
recip_rank              all     1.0000
iprec_at_recall_0.00    all     1.0000
iprec_at_recall_0.10    all     1.0000
iprec_at_recall_0.20    all     0.9333
iprec_at_recall_0.30    all     0.8847
iprec_at_recall_0.40    all     0.8698
iprec_at_recall_0.50    all     0.7291
iprec_at_recall_0.60    all     0.6147
iprec_at_recall_0.70    all     0.5322
iprec_at_recall_0.80    all     0.3667
iprec_at_recall_0.90    all     0.2815
iprec_at_recall_1.00    all     0.2815
P_5                     all     0.8400
P_10                    all     0.6733
P_15                    all     0.5244
P_20                    all     0.4033
P_30                    all     0.2689
P_100                   all     0.0807
P_200                   all     0.0403
P_500                   all     0.0161
P_1000                  all     0.0081
```

MODEL	Initial MAP score	Final MAP score
BM25	0.6756	0.6809
VSM	0.13(default value)	0.6756