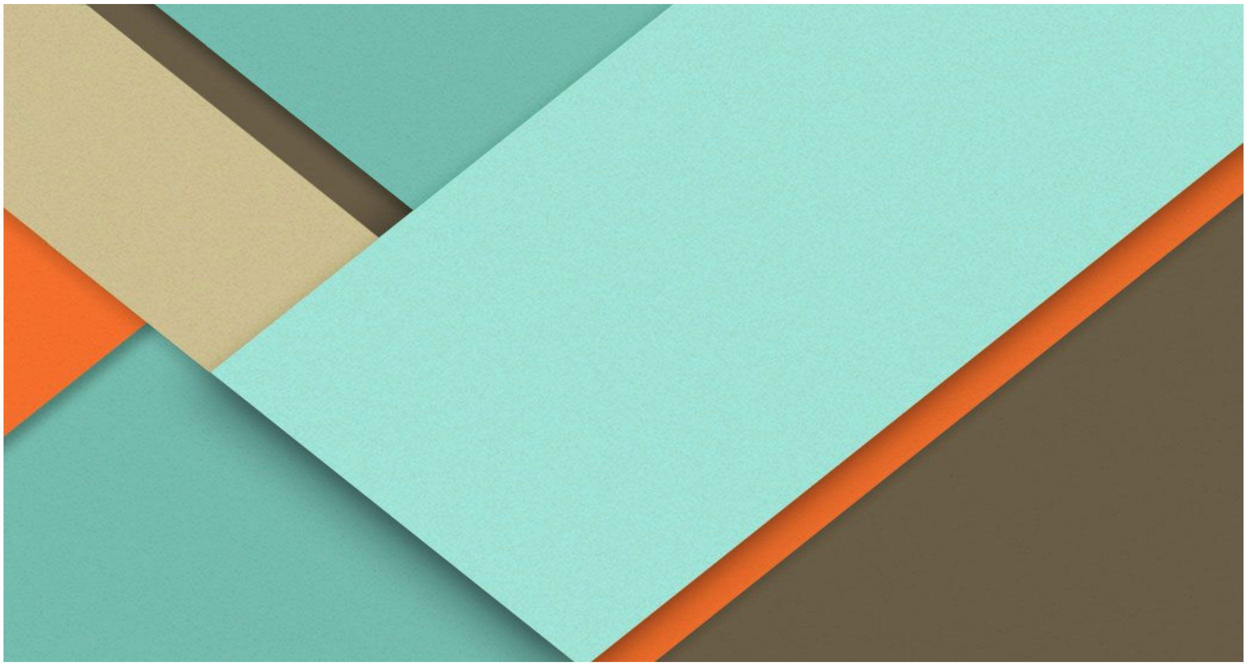
CSE 535

Project 3 : Evaluation of IR Models

Information Retrieval



Group No. : 47

NAINA NIGAM (50208030)

VANSHIKA NIGAM (50208031)

November 10th, 2016

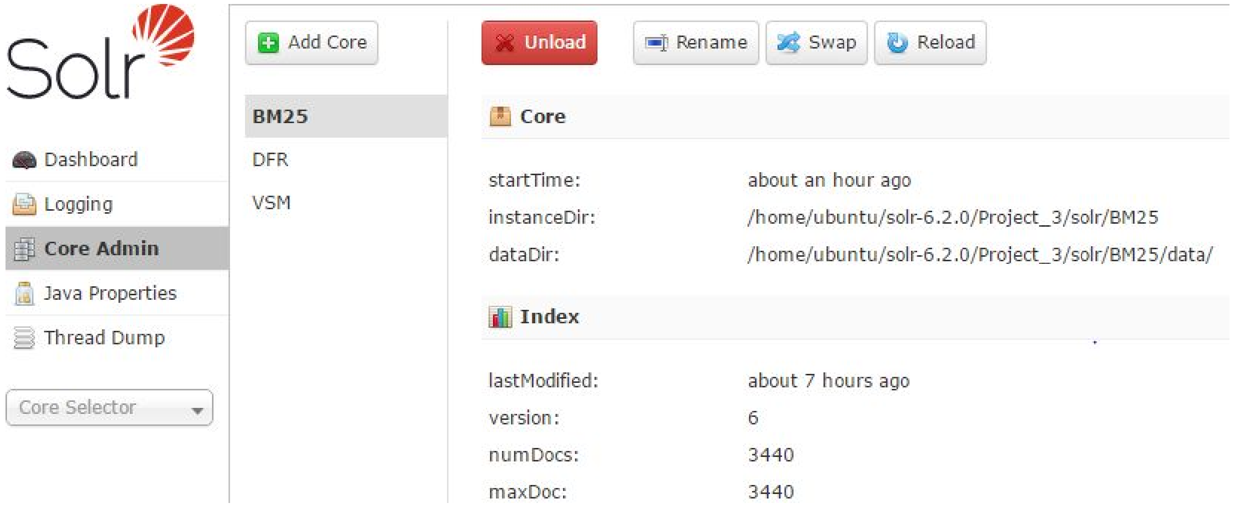
OVERVIEW

The objective of this IR project is to implement the three IR models namely Vector Space Model, BM25 and Divergence from Randomness model using Solr, evaluate the IR system and improve the search result based on the understanding of the models.

- **BM25 :** The Best Matching (BM25) algorithm is a probabilistic Information Retrieval (IR) model. Its often called Okapi Weighting, which is a ranking function in Information Retrieval used by search engines to rank matching documents according to their relevance to given search query.
- **Vector Space Model :** Also known as Term Vector Model, is an algebraic model used for representing documents and queries as vectors in the term space. It allows computing a continuous degree of similarity between queries and documents, ranking documents according to their possible relevance, partial matching.
- **Divergence from Randomness :** The Divergence from Randomness (DFR) paradigm is a generalisation of one of the very first models of Information Retrieval, Harter's 2-Poisson indexing-model. It is based on the following components : Randomness Model, First Normalization and Term Frequency Normalization.

We successfully implemented the 3 models and created 3 cores for the same. Changes were made as we made progress on improvisation of the model

Please see the snapshot of the Solr UI displaying the 3 core named according to their models.



startTime: about an hour ago
instanceDir: /home/ubuntu/solr-6.2.0/Project_3/solr/BM25
dataDir: /home/ubuntu/solr-6.2.0/Project_3/solr/BM25/data/

lastModified: about 7 hours ago
version: 6
numDocs: 3440
maxDoc: 3440

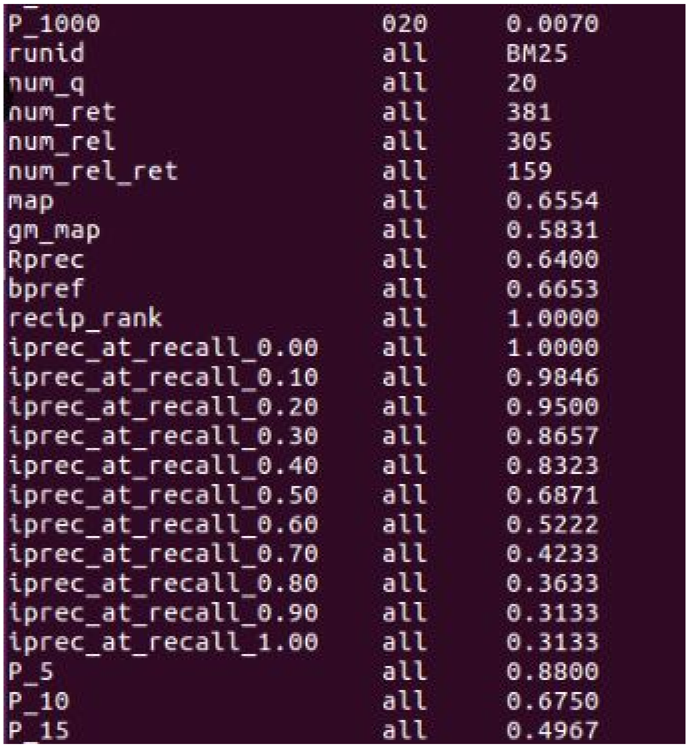
BM25 Model

Initial Implementation : This is the default model. In order to implement the model changes were made in the schema.xml file by inserting the following statement.

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

On calculating the MAP value through trec_eval the result received was **0.6554.**

Please find below the screenshot for the same:



```
P_1000                  020     0.0070
runid                   all     BM25
num_q                   all     20
num_ret                 all     381
num_rel                 all     305
num_rel_ret             all     159
map                     all     0.6554
gm_map                  all     0.5831
Rprec                   all     0.6400
bpref                   all     0.6653
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     0.9500
iprec_at_recall_0.30    all     0.8657
iprec_at_recall_0.40    all     0.8323
iprec_at_recall_0.50    all     0.6871
iprec_at_recall_0.60    all     0.5222
iprec_at_recall_0.70    all     0.4233
iprec_at_recall_0.80    all     0.3633
iprec_at_recall_0.90    all     0.3133
iprec_at_recall_1.00    all     0.3133
P_5                     all     0.8800
P_10                    all     0.6750
P_15                    all     0.4967
```

Improvement : In order to get a better MAP value , we played around the values of the solr.BM25SimilarityFactory class in order to get the improved score.

K_1: This parameter controls how quickly an increase in term frequency results in term-frequency saturation. **The default value is 1.2** Lower values result in quicker saturation, and higher values in slower saturation.

b: This parameter controls how much effect field-length normalization should have. **The default is 0.75**.

Some of the value pair used in order to end up to a final value

- (k_1,b)=(1.0,1.0) =>MAP=0.6363
- (k_1,b)=(1.0,0.7) =>MAP=0.6563
- (k_1,b)=(1.25,0.7) => MAP=0.6573

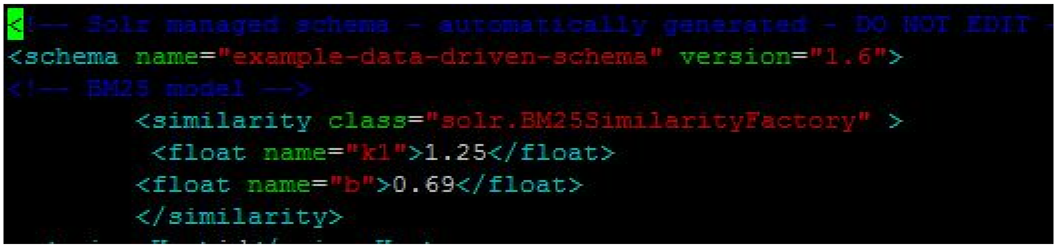
Final value:

k_1=1.25

b=0.69

MAP =0.6871

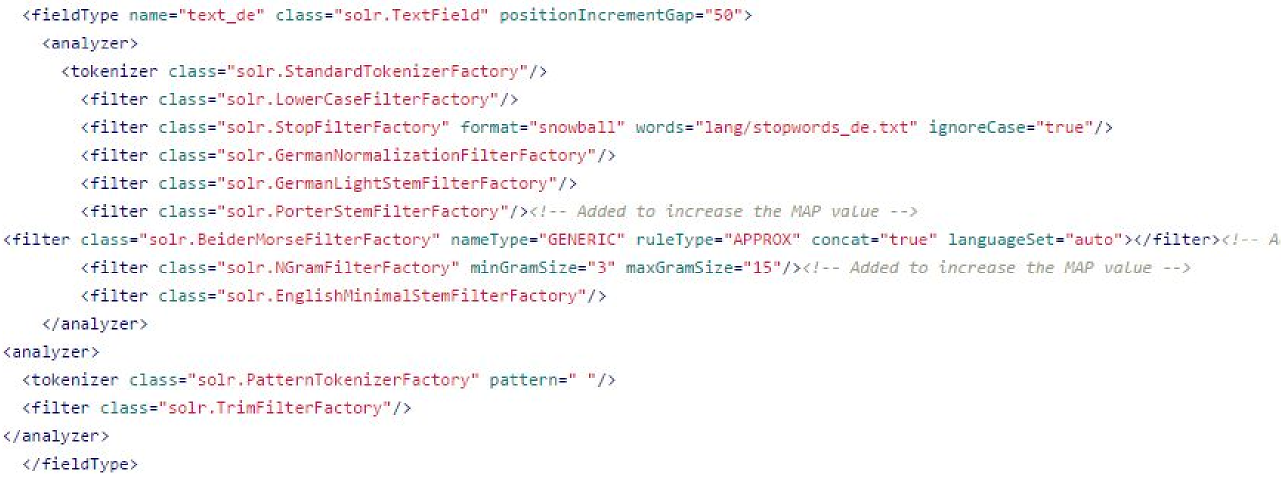
Below are the changes in the schema.xml file



```
<!-- Solr managed schema - automatically generated - DO NOT EDIT -
<schema name="example-data-driven-schema" version="1.6">
<!-- BM25 model -->
        <similarity class="solr.BM25SimilarityFactory" >
         <float name="k1">1.25</float>
        <float name="b">0.69</float>
        </similarity>
```

On making further changes in values of b and k_1 there is negligible change in the MAP value so we decided to make changes in the **schema.xml** file by adding those filter classes that would enhance the performance and in turn the MAP value.

Please see the screenshot below :

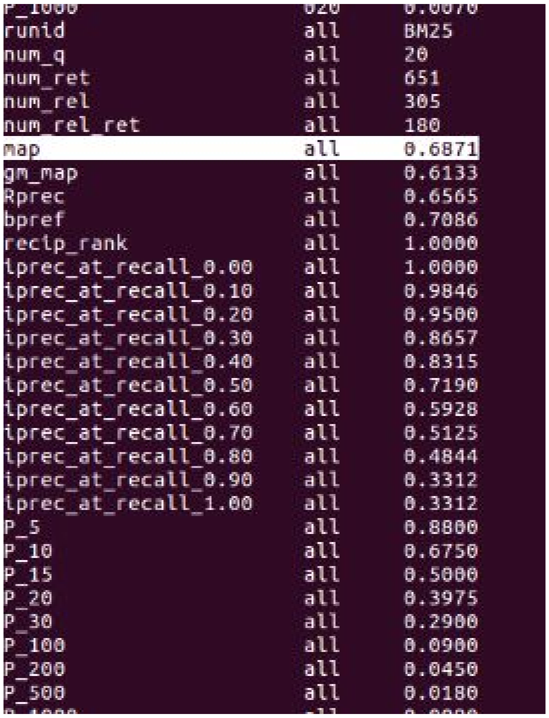


```
  <fieldType name="text_de" class="solr.TextField" positionIncrementGap="50">
    <analyzer>
      <tokenizer class="solr.StandardTokenizerFactory"/>
        <filter class="solr.LowerCaseFilterFactory"/>
        <filter class="solr.StopFilterFactory" format="snowball" words="lang/stopwords_de.txt" ignoreCase="true"/>
        <filter class="solr.GermanNormalizationFilterFactory"/>
        <filter class="solr.GermanLightStemFilterFactory"/>
        <filter class="solr.PorterStemFilterFactory"/><!-- Added to increase the MAP value -->
<filter class="solr.BeiderMorseFilterFactory" nameType="GENERIC" ruleType="APPROX" concat="true" languageSet="auto"></filter><!-- A
        <filter class="solr.NGramFilterFactory" minGramSize="3" maxGramSize="15"/><!-- Added to increase the MAP value -->
        <filter class="solr.EnglishMinimalStemFilterFactory"/>
    </analyzer>
<analyzer>
  <tokenizer class="solr.PatternTokenizerFactory" pattern=" "/>
  <filter class="solr.TrimFilterFactory"/>
</analyzer>
  </fieldType>
```

We have added following Filter classes in schema.xml:

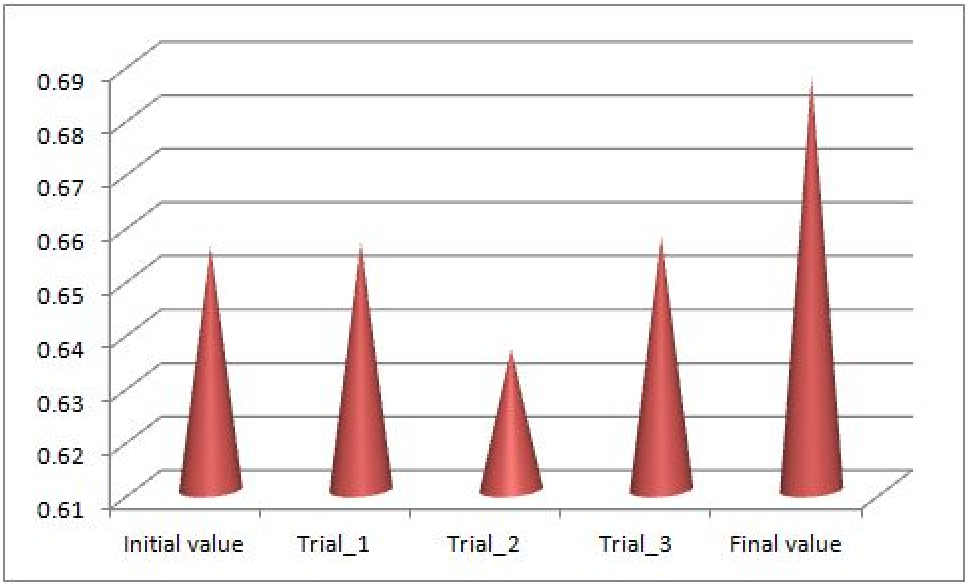
- **Porter Stem Filter** - This filter applies Porter Stem Algorithm for English, it returns the root word while dropping the suffix(es). It has been benchmarked as four times faster than the English Snowball stemmer, so can provide a performance enhancement.
- **Beider Morse Filter** - This filter allows identification of similar names, even if they are spelled differently or in different languages.
- **N Gram Filter** - It generates n-gram tokens of sizes in the given range of minGramSize and maxGramSize.

After doing the above changes we successfully landed to an improved MAP value.



```
P_1000                  020     0.0070
runid                   all     BM25
num_q                   all     20
num_ret                 all     651
num_rel                 all     305
num_rel_ret             all     180
map                     all     0.6871
gm_map                  all     0.6133
Rprec                   all     0.6565
bpref                   all     0.7086
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     0.9500
iprec_at_recall_0.30    all     0.8657
iprec_at_recall_0.40    all     0.8315
iprec_at_recall_0.50    all     0.7190
iprec_at_recall_0.60    all     0.5928
iprec_at_recall_0.70    all     0.5125
iprec_at_recall_0.80    all     0.4844
iprec_at_recall_0.90    all     0.3312
iprec_at_recall_1.00    all     0.3312
P_5                     all     0.8800
P_10                    all     0.6750
P_15                    all     0.5000
P_20                    all     0.3975
P_30                    all     0.2900
P_100                   all     0.0900
P_200                   all     0.0450
P_500                   all     0.0180
P_1000                  all     0.0090
```

The **Graphical Analysis of BM25 model** is :



Divergence From Randomness (DFR model)

There are three components of DFR (strings).

Initial Implementation: To implement this model in the default mode , a similarity class was added to the schema.xml file as under with the default values with respect to the information given in the project guidelines.

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

On calculating the MAP value through trec_eval the result received was **0.6468**

Improvement : In order to increase the map score , there was need to change the values of the 3 strings.

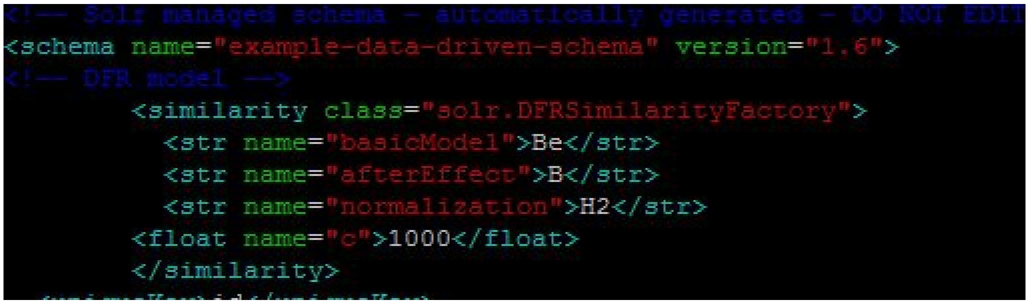
We have to put in the right combination as much possible of **basicModel** ,**afterEffect** and **normalization** in order to improve the score.

We finally settled for the following :

1.basicModel:**Be**: Limiting form of Bose-Einstein

2. afterEffect:**B**: Ratio of two Bernoulli processes

3.normalization: **H2**: term frequency density inversely related to length



```
<!-- Solr managed schema - automatically generated - DO NOT EDIT
<schema name="example-data-driven-schema" version="1.6">
<!-- DFR model -->
        <similarity class="solr.DFRSimilarityFactory">
          <str name="basicModel">Be</str>
          <str name="afterEffect">B</str>
          <str name="normalization">H2</str>
        <float name="c">1000</float>
        </similarity>
```

With Improved MAP value =**0.6965**

Some of the trial values used in order to end up to a final value were :

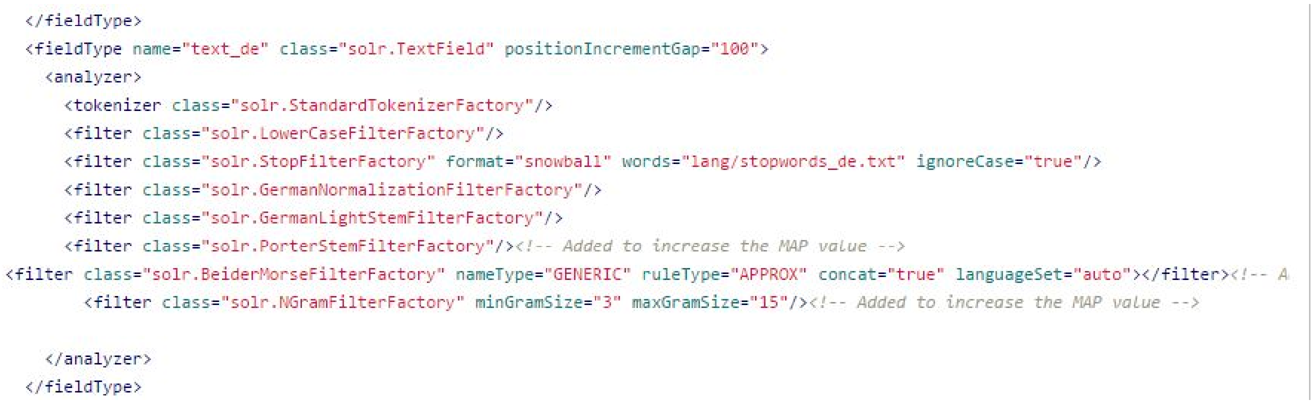
➔ (basicModel , afterEffect ,normalization) => (D: Divergence approximation of the Binomial,B: Ratio of two Bernoulli processes, H2: term frequency density inversely related to length) MAP =0.6822

➔ (basicModel , afterEffect ,normalization) =>(P: Poisson approximation of the Binomial, B: Ratio of two Bernoulli processes,H3:term frequency normalization provided by Dirichlet prior) MAP=0.6420

➔ (basicModel , afterEffect ,normalization)=> (Be: Limiting form of Bose-Einstein,L: Laplace's law of succession,H2: term frequency density inversely related to length) MAP=0.6894

Along with the additions in schema.xml (similar to that for BM25 model) we also modified the query and added The DisMax query parser to improve the score. It is designed to process simple phrases entered by users and to search for individual terms across several fields using different weighting (boosts) based on the significance of each field.

http://54.68.99.12:8983/solr/DFR/select?q=%27+qText+%27&fl=id%2Cscore&wt=json&indent=true &rows=20&**defType=dismax**

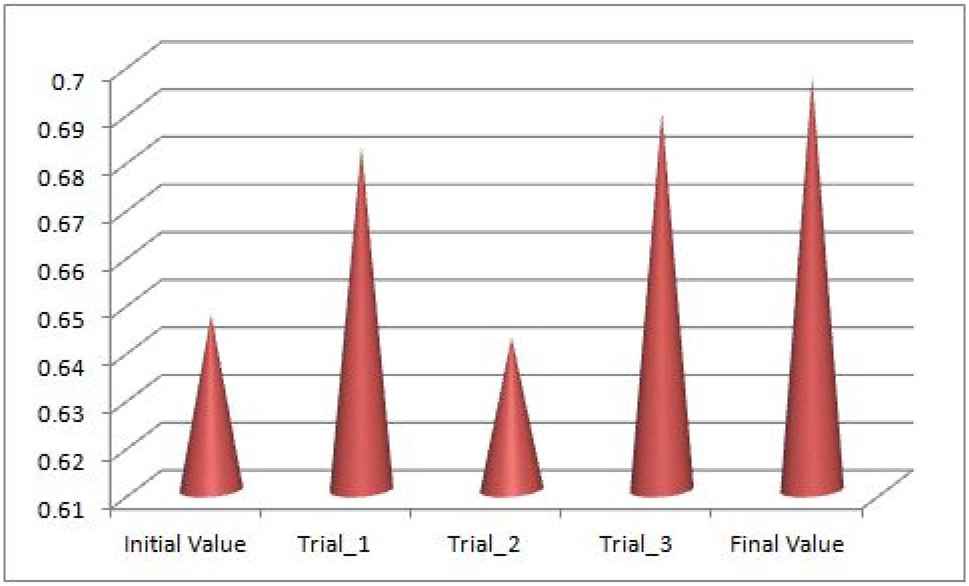


```
    </fieldType>
    <fieldType name="text_de" class="solr.TextField" positionIncrementGap="100">
      <analyzer>
        <tokenizer class="solr.StandardTokenizerFactory"/>
        <filter class="solr.LowerCaseFilterFactory"/>
        <filter class="solr.StopFilterFactory" format="snowball" words="lang/stopwords_de.txt" ignoreCase="true"/>
        <filter class="solr.GermanNormalizationFilterFactory"/>
        <filter class="solr.GermanLightStemFilterFactory"/>
        <filter class="solr.PorterStemFilterFactory"/><!-- Added to increase the MAP value -->
  <filter class="solr.BeiderMorseFilterFactory" nameType="GENERIC" ruleType="APPROX" concat="true" languageSet="auto"></filter><!-- A
          <filter class="solr.NGramFilterFactory" minGramSize="3" maxGramSize="15"/><!-- Added to increase the MAP value -->

      </analyzer>
    </fieldType>
```

Similar changes were made to text_en and text_ru.

The Graphical Analysis of DFR model is :



Vector Space Model(VSM)

The vector space model procedure can be divided into three stages.

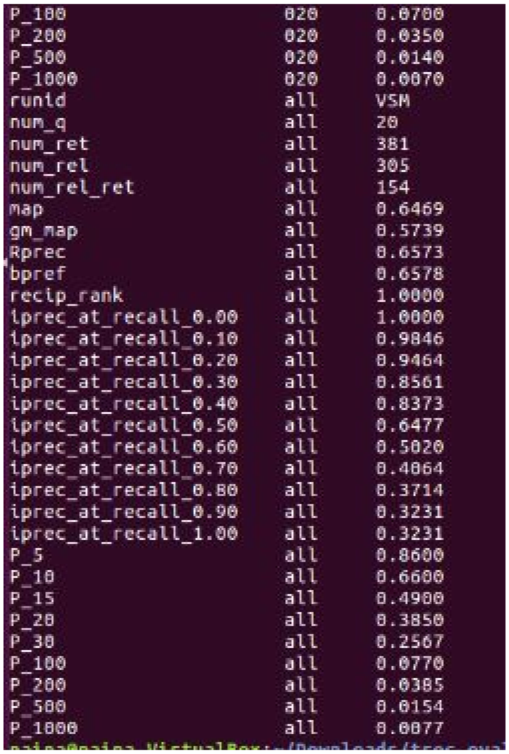
The first stage is the document indexing where content bearing terms are extracted from the document text.

The second stage is the weighting of the indexed terms to enhance retrieval of document relevant to the user.

The last stage ranks the document with respect to the query according to a similarity measure.

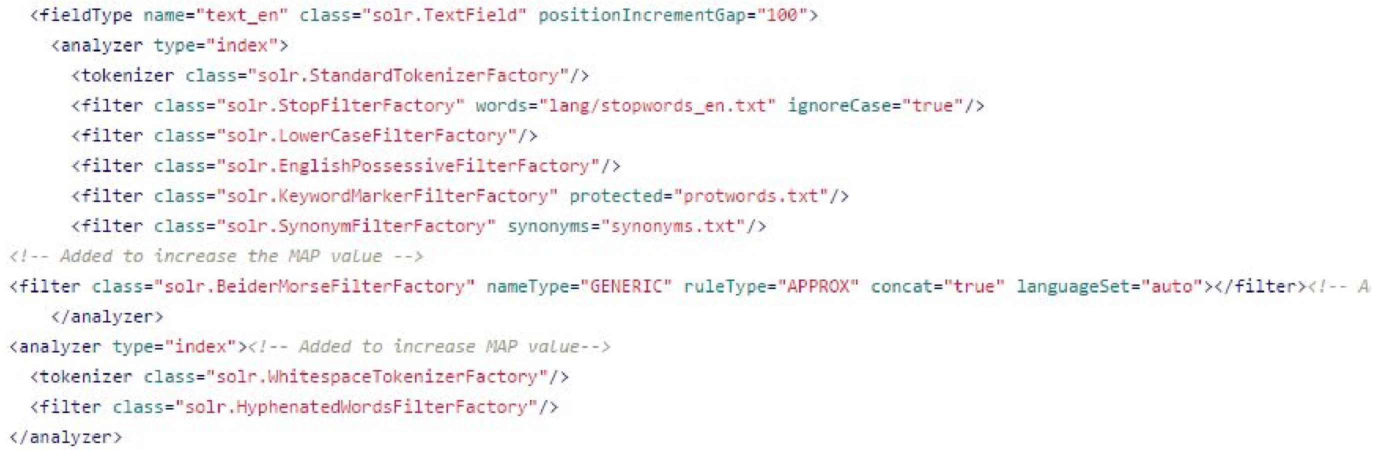
Similarity Class for VSM : **ClassicSimilarityFactory** was added to schema.xml

On running the default settings for VSM the result obtained for MAP value was **0.6469.**



```
P_100                   020     0.0700
P_200                   020     0.0350
P_500                   020     0.0140
P_1000                  020     0.0070
runid                   all     VSM
num_q                   all     20
num_ret                 all     381
num_rel                 all     305
num_rel_ret             all     154
map                     all     0.6469
gm_map                  all     0.5739
Rprec                   all     0.6573
bpref                   all     0.6578
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     0.9464
iprec_at_recall_0.30    all     0.8561
iprec_at_recall_0.40    all     0.8373
iprec_at_recall_0.50    all     0.6477
iprec_at_recall_0.60    all     0.5020
iprec_at_recall_0.70    all     0.4064
iprec_at_recall_0.80    all     0.3714
iprec_at_recall_0.90    all     0.3231
iprec_at_recall_1.00    all     0.3231
P_5                     all     0.8600
P_10                    all     0.6600
P_15                    all     0.4900
P_20                    all     0.3850
P_30                    all     0.2567
P_100                   all     0.0770
P_200                   all     0.0385
P_500                   all     0.0154
P_1000                  all     0.0077
```

In order to improve the performance of this model we added some more stopwords to **stopwords_en.txt**, **stopwords_de.txt**, **stopwords_ru.txt** so as to get better precision and recall values. Also we added a list of synonyms of common words to **synonyms.txt** so as to get better MAP result . These two changes can be seen in the screenshot below:



```
  <fieldType name="text_en" class="solr.TextField" positionIncrementGap="100">
    <analyzer type="index">
      <tokenizer class="solr.StandardTokenizerFactory"/>
      <filter class="solr.StopFilterFactory" words="lang/stopwords_en.txt" ignoreCase="true"/>
      <filter class="solr.LowerCaseFilterFactory"/>
      <filter class="solr.EnglishPossessiveFilterFactory"/>
      <filter class="solr.KeywordMarkerFilterFactory" protected="protwords.txt"/>
      <filter class="solr.SynonymFilterFactory" synonyms="synonyms.txt"/>
<!-- Added to increase the MAP value -->
<filter class="solr.BeiderMorseFilterFactory" nameType="GENERIC" ruleType="APPROX" concat="true" languageSet="auto"></filter><!-- A
    </analyzer>
<analyzer type="index"><!-- Added to increase MAP value-->
  <tokenizer class="solr.WhitespaceTokenizerFactory"/>
  <filter class="solr.HyphenatedWordsFilterFactory"/>
</analyzer>
```

Following Filter classes have been added:

- **Synonym Filter -** Each token is looked up in the list of synonyms and if a match is found, then the synonym is emitted in place of the token. The position value of the new tokens are set such they all occur at the same position as the original token.

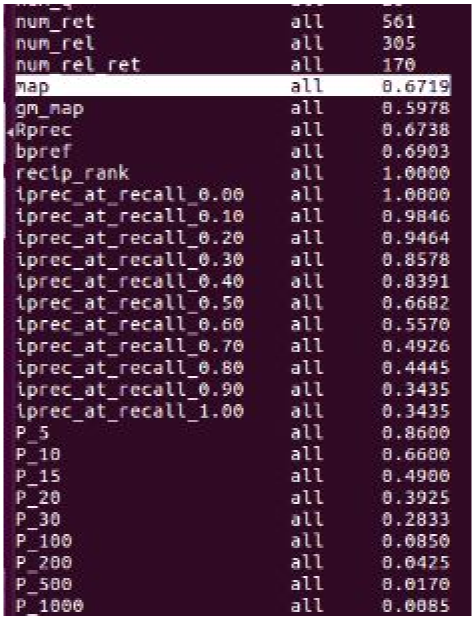
- **Hyphenated Words Filter** - This filter is used to use hyphenated terms as one single terms in order to provide better matching results at the time of query searching and matching.

  Both these filters have been added to provide better MAP results in the VSM model and it is done so by performing better matching of terms in the document with the query terms.

Before arriving to the final MAP value we had some intermediated values as

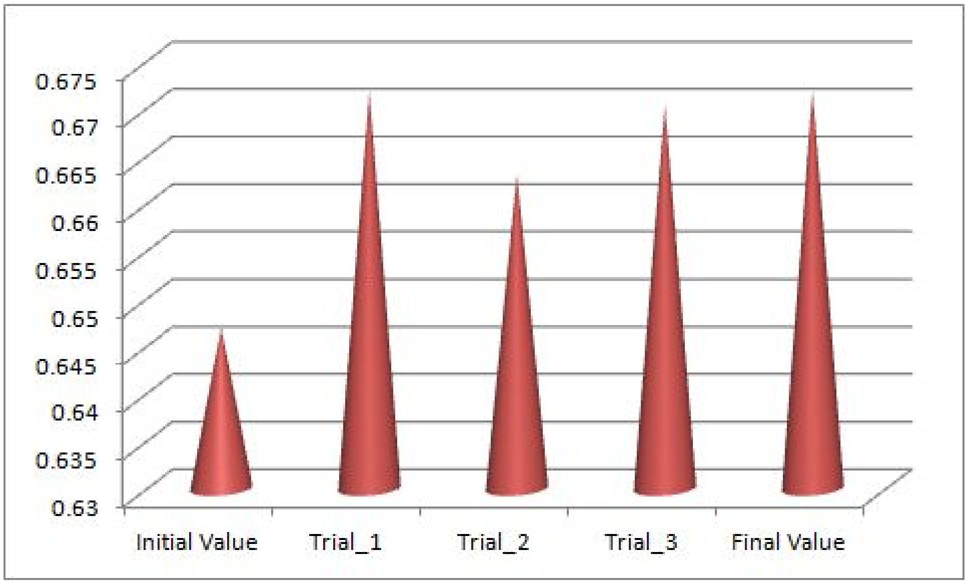
- ➔ MAP =0.6728
- ➔ MAP=0.6631
- ➔ MAP=0.6704

The final improved MAP value is **0.6719** which can be seen in the screenshot below:

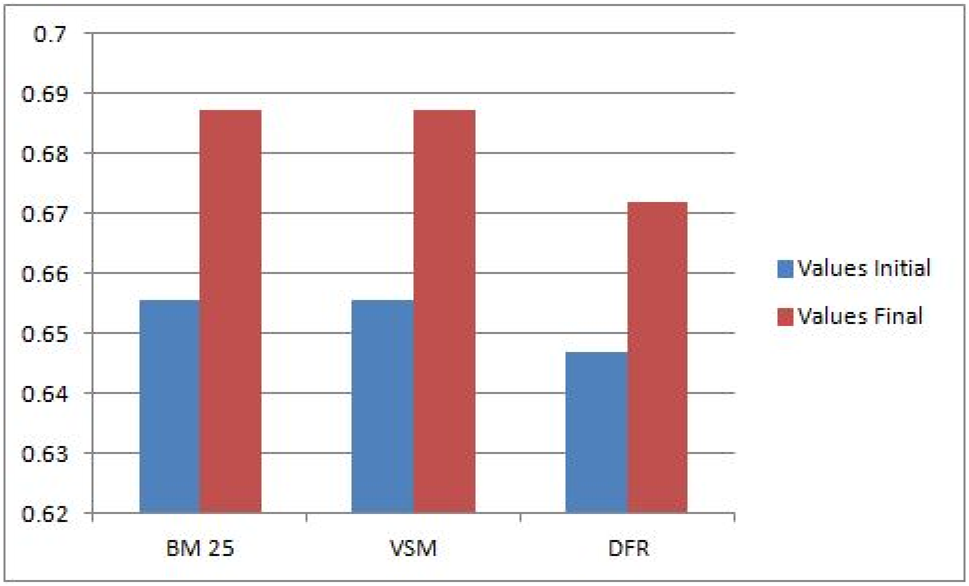


```
num_ret                 all     561
num_rel                 all     305
num_rel_ret             all     170
map                     all     0.6719
gm_map                  all     0.5978
Rprec                   all     0.6738
bpref                   all     0.6903
recip_rank              all     1.0000
iprec_at_recall_0.00    all     1.0000
iprec_at_recall_0.10    all     0.9846
iprec_at_recall_0.20    all     0.9464
iprec_at_recall_0.30    all     0.8578
iprec_at_recall_0.40    all     0.8391
iprec_at_recall_0.50    all     0.6682
iprec_at_recall_0.60    all     0.5570
iprec_at_recall_0.70    all     0.4926
iprec_at_recall_0.80    all     0.4445
iprec_at_recall_0.90    all     0.3435
iprec_at_recall_1.00    all     0.3435
P_5                     all     0.8600
P_10                    all     0.6600
P_15                    all     0.4900
P_20                    all     0.3925
P_30                    all     0.2833
P_100                   all     0.0850
P_200                   all     0.0425
P_500                   all     0.0170
P_1000                  all     0.0085
```

The Graphical Analysis of VSM model is :



Collective Graphical Analysis of the three models



Conclusion

The three IR models have been implemented successfully and we tried to improve the performance of all the three models by using their properties and the features provided by Solr 6.2.0.

In BM25 model we settled for values of b and k_1 as 0.69 and 1.25 respectively, as better results were achieved through these values and also by adding N gram filter, Beider Morse filter and Porter Stem Filter.

In DFR model with the help of the 3 components we improved the MAP value along with some changes in the schema.xml.

In VSM model we added Synonym Filter , Hyphenated Filter as well added more stopwords to the stopwords.txt file of all three language fields.

All these provided some improvements in the system but there is a lot more scope of improvement to do in future prospects.