CSE 435/535 Information Retrieval (Fall 2016)

Project 3: Evaluation of IR models Report

Prepared by:

ANURAG DEVULAPALLI – ANURAGDE

VIPIN KUMAR - VKUMAR25

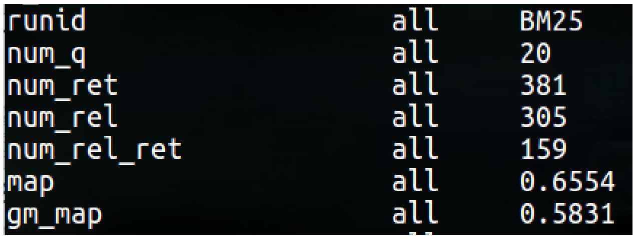
Table of Contents

A. Implementation of BM25 Model in Solr 6.2.0

I. Steps to implement with default settings.

Solr 6.2.0 implements BM25 model by default.

MAP Value: 0.6554

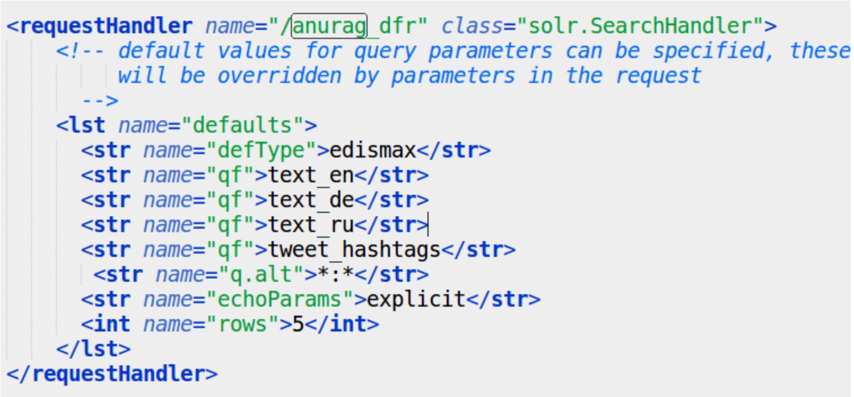


```
runid                   all     BM25
num_q                   all     20
num_ret                 all     381
num_rel                 all     305
num_rel_ret             all     159
map                     all     0.6554
gm_map                  all     0.5831
```

II. Steps taken to improve performance.

1. Changed the default search field.

We created a new request handler for search and defined the query fields to be searched as below:

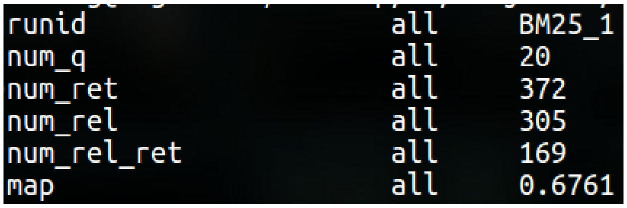


```
<requestHandler name="/anurag_dfr" class="solr.SearchHandler">
    <!-- default values for query parameters can be specified, these
         will be overridden by parameters in the request
      -->
    <lst name="defaults">
      <str name="defType">edismax</str>
      <str name="qf">text_en</str>
      <str name="qf">text_de</str>
      <str name="qf">text_ru</str>
      <str name="qf">tweet_hashtags</str>
       <str name="q.alt">*:*</str>
      <str name="echoParams">explicit</str>
      <int name="rows">5</int>
    </lst>
</requestHandler>
```

Reason: Solr's default search field is _text_ which is included in the '/select' request handler. In order to search the query over all the fields we added a new request handler.

Result: Success

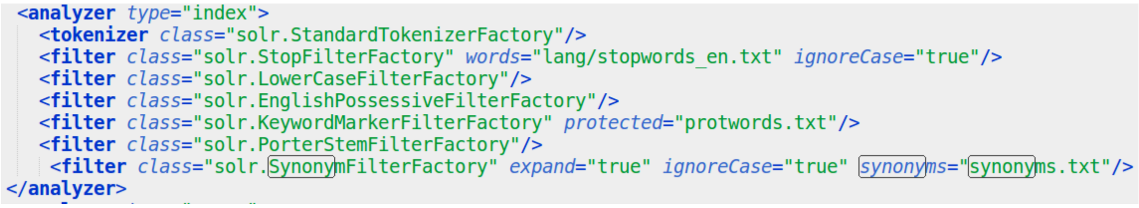
New Map: 0.6761



```
runid                   all     BM25_1
num_q                   all     20
num_ret                 all     372
num_rel                 all     305
num_rel_ret             all     169
map                     all     0.6761
```

2. Query expansion by multilingual search.

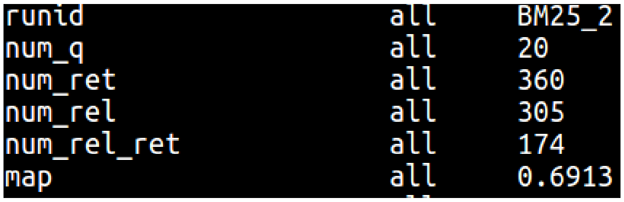
Performed the synonym filtering at Index time for multilingual search as below:



```
 <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.PorterStemFilterFactory"/>
    <filter class="solr.SynonymFilterFactory" expand="true" ignoreCase="true" synonyms="synonyms.txt"/>
</analyzer>
```

Reason: To achieve multilingual search by translating the *important* (ex: nouns) words from the queries and added to synonyms.txt file and applied the filter at Index time. Reason for translating only the important words is to give more weightage for these words.

Result: Success
New Map: 0.6913

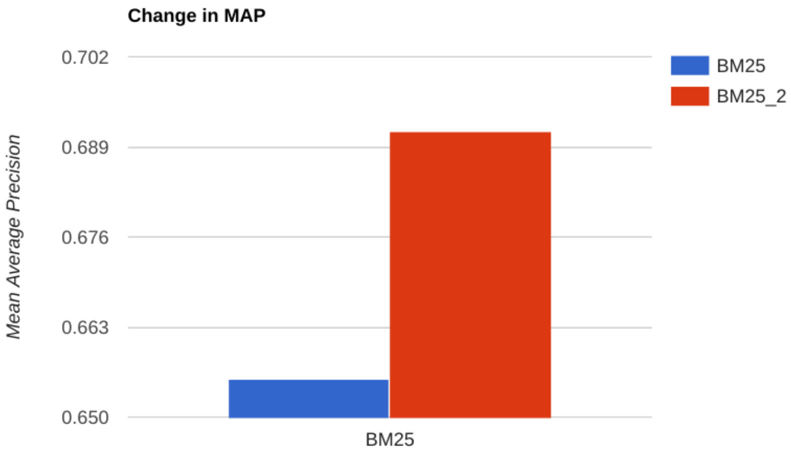


```
runid                   all     BM25_2
num_q                   all     20
num_ret                 all     360
num_rel                 all     305
num_rel_ret             all     174
map                     all     0.6913
```

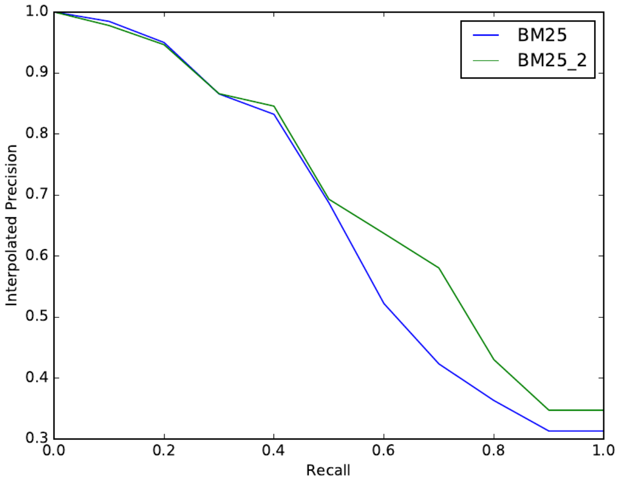
Results:

Initial MAP: 0.6554

Final MAP: 0.6913



Comparison of MAP before and after



Interpolated precision vs recall comparison before and after modification

III. Unsuccessful results:

We implemented some of the following but, it reduced the overall map :

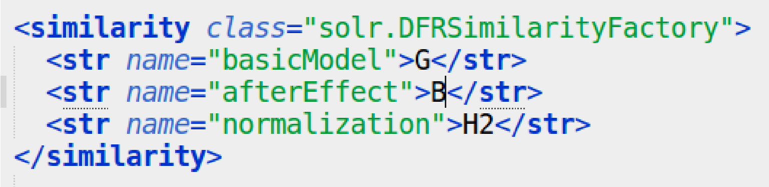
- Modified the k1 and b values.
- Installed a plugin[1] for synonym expansion at query time,which gives more weightage to original word than synonyms for better relevancy.
- Tried to remove the near duplicates terms (using facet parameters in query) from the search results, as some of the tweets have almost same content with different id’s.
- Assigned higher weightage to some query fields such as text_en,if the query language is in English, so that the results retrieved first are from the queried language.

[1] https://github.com/healthonnet/hon-lucene-synonyms

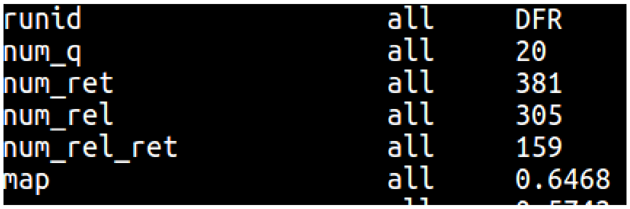
B. Implementation of DFR Model in Solr 6.2.0

I. Steps to implement with default settings

We need to add the following code to schema.xml.



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



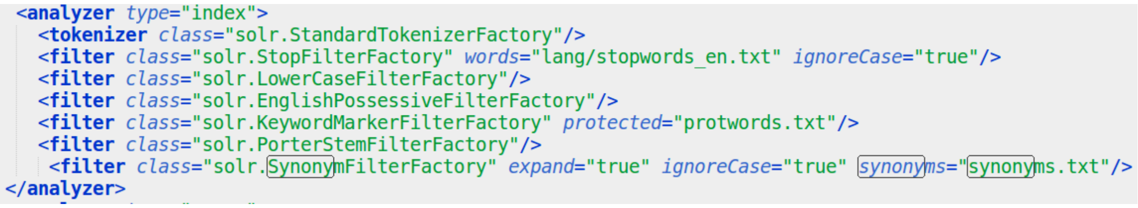
```
runid                   all     DFR
num_q                   all     20
num_ret                 all     381
num_rel                 all     305
num_rel_ret             all     159
map                     all     0.6468
```

Initial MAP : 0.6468

II. Steps taken to improve performance

1. Query expansion by multilingual search.

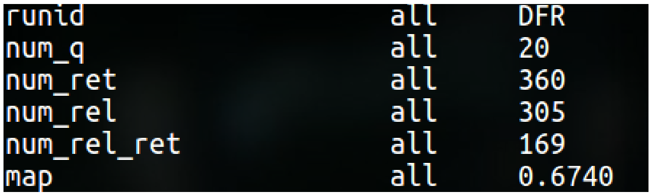
Performed the synonym filtering at Index time for multilingual search as below:



```
 <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.PorterStemFilterFactory"/>
    <filter class="solr.SynonymFilterFactory" expand="true" ignoreCase="true" synonyms="synonyms.txt"/>
</analyzer>
```

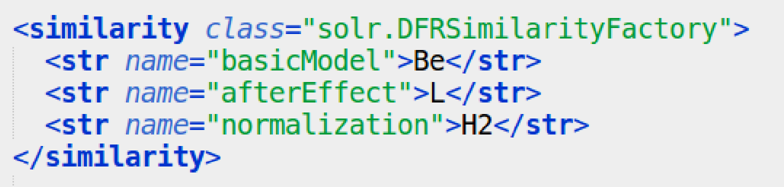
Reason: To achieve multilingual search by translating the *important* (ex: nouns) words from the queries and added to synonyms.txt file and applied the filter at Index time. Reason for translating only the important words is to give more weightage for these words.

Result: Success
New Map: 0.6740



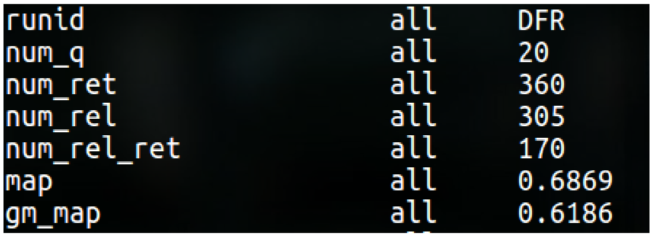
```
runid                   all     DFR
num_q                   all     20
num_ret                 all     360
num_rel                 all     305
num_rel_ret             all     169
map                     all     0.6740
```

2. Modified the basic model and after effect parameters.



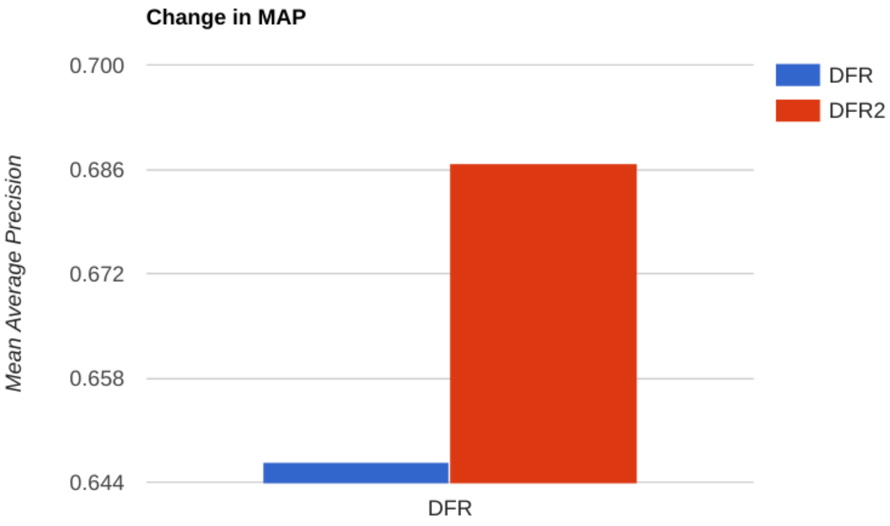
```
<similarity class="solr.DFRSimilarityFactory">
  <str name="basicModel">Be</str>
  <str name="afterEffect">L</str>
  <str name="normalization">H2</str>
</similarity>
```

Result: Success
New Map: 0.6869

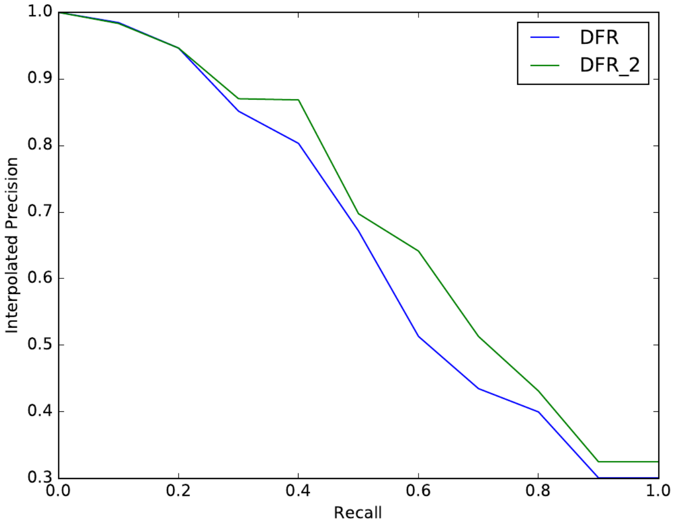


```
runid                   all     DFR
num_q                   all     20
num_ret                 all     360
num_rel                 all     305
num_rel_ret             all     170
map                     all     0.6869
gm_map                  all     0.6186
```

Results:
Initial MAP : 0.6468
Final MAP : 0.6869



Comparison of MAP before and after



Interpolated precision vs recall comparison before and after modification

III. Unsuccessful results:

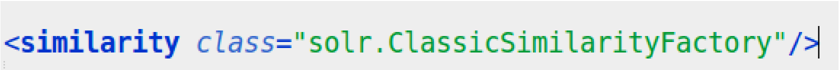
We implemented some of the following but, it reduced the overall map :

- Modified the c value.
- Installed a plugin for synonym expansion at query time,which gives more weightage to original word than synonyms for better relevancy.
- Tried to remove the near duplicates terms (using facet parameters in query) from the search results, as some of the tweets have almost same content with different id's.
- Assigned higher weightage to some query fields such as text_en,if the query language is in English, so that the results retrieved first are from the queried language.

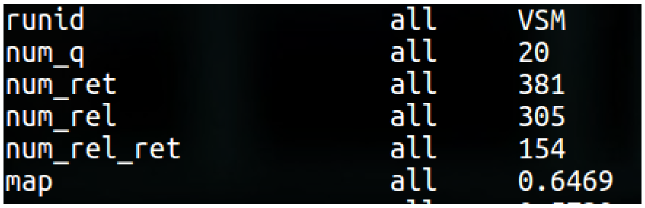
C. Implementation of Vector Space Model

I. Steps to implement with default settings.

We need to add the following code to schema.xml.



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



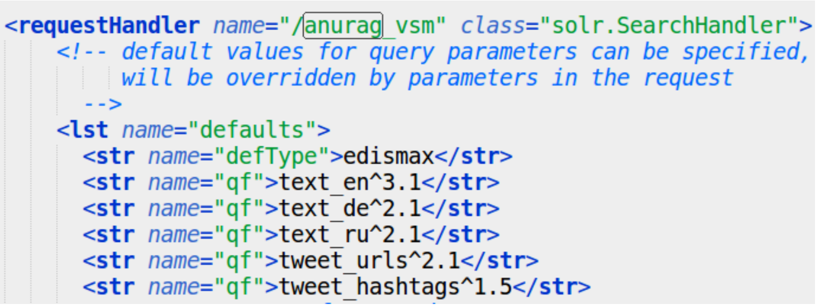
```
runid                   all     VSM
num_q                   all     20
num_ret                 all     381
num_rel                 all     305
num_rel_ret             all     154
map                     all     0.6469
```

Initial MAP : 0.6469

II. Steps taken to improve performance.

1. Changed the default search field.

We created a new request handler for search and defined the query fields to be searched as below:

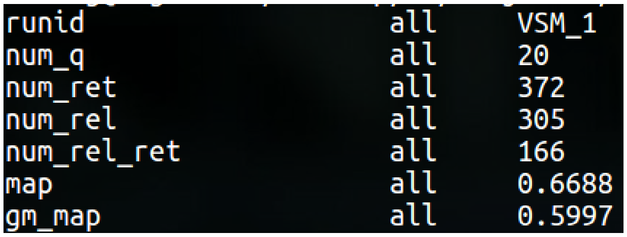


```
<requestHandler name="/anurag_vsm" class="solr.SearchHandler">
    <!-- default values for query parameters can be specified,
         will be overridden by parameters in the request
      -->
    <lst name="defaults">
      <str name="defType">edismax</str>
      <str name="qf">text_en^3.1</str>
      <str name="qf">text_de^2.1</str>
      <str name="qf">text_ru^2.1</str>
      <str name="qf">tweet_urls^2.1</str>
      <str name="qf">tweet_hashtags^1.5</str>
```

Reason: Solr's default search field is _text_ which is included in the '/select' request handler. In order to search the query over all the fields with different weights we added a new request handler.

Result: Success

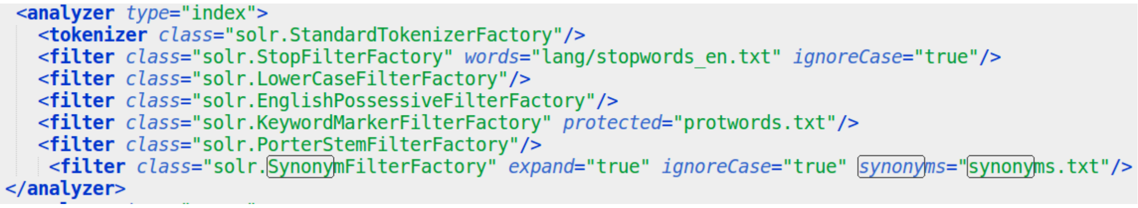
New Map: 0.6688



```
runid                   all     VSM_1
num_q                   all     20
num_ret                 all     372
num_rel                 all     305
num_rel_ret             all     166
map                     all     0.6688
gm_map                  all     0.5997
```

2. Query expansion by multilingual search.

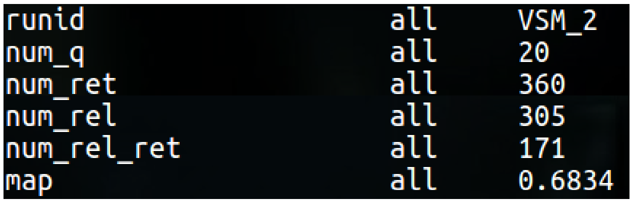
Performed the synonym filtering at Index time for multilingual search as below:



```
 <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.PorterStemFilterFactory"/>
    <filter class="solr.SynonymFilterFactory" expand="true" ignoreCase="true" synonyms="synonyms.txt"/>
</analyzer>
```

Reason: To achieve multilingual search by translating the *important* (ex: nouns) words from the queries and added to synonyms.txt file and applied the filter at Index time. Reason for translating only the important words is to give more weightage for these words.

Result: Success
New Map: 0.6834

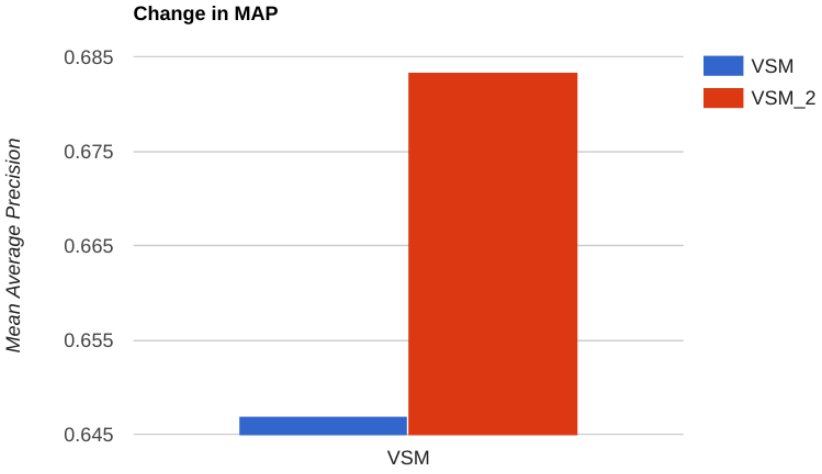


```
runid                   all     VSM_2
num_q                   all     20
num_ret                 all     360
num_rel                 all     305
num_rel_ret             all     171
map                     all     0.6834
```

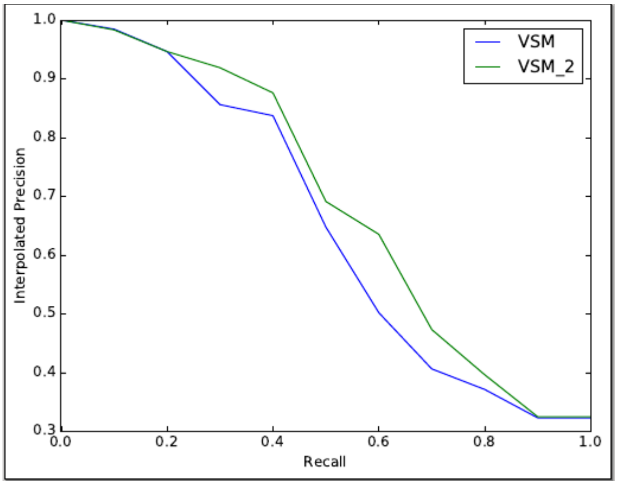
Results:

Initial MAP: 0.6469

Final MAP: 0.6834



Comparison of MAP before and after



Interpolated precision vs recall comparison before and after modification

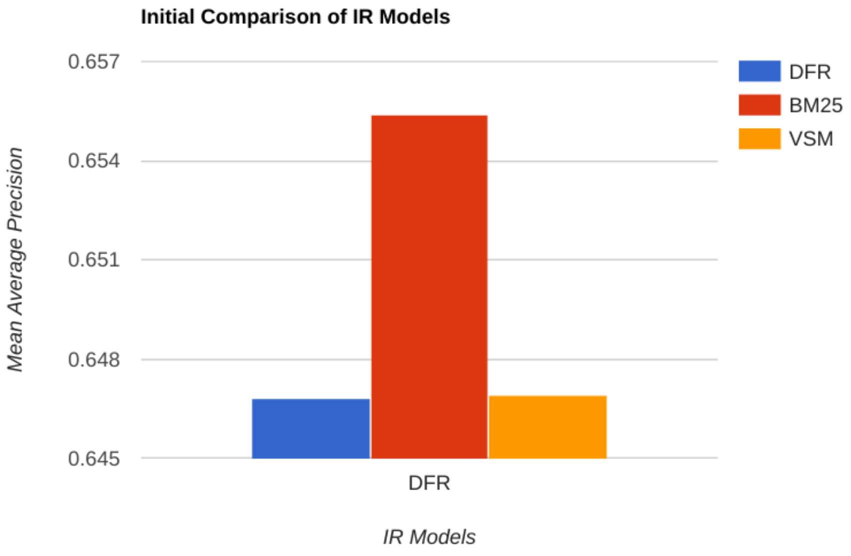
III. Unsuccessful results:

We implemented some of the following but, it reduced the overall map :

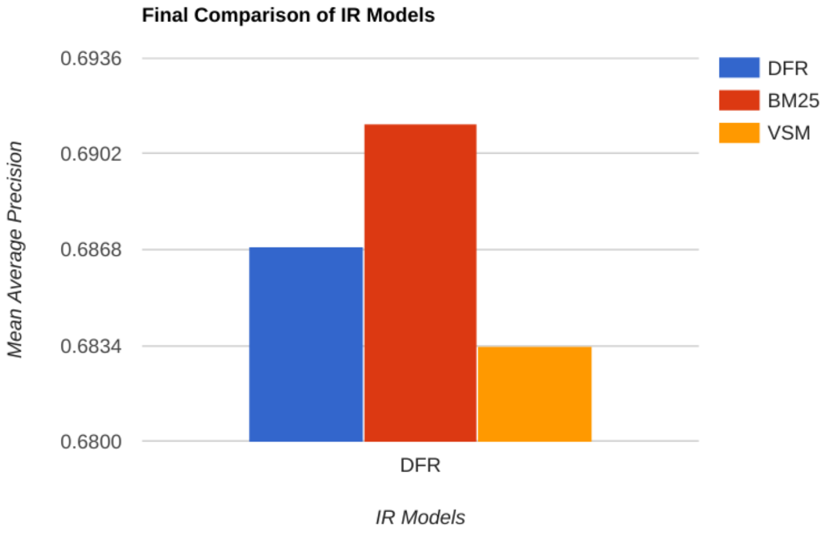
- Implemented the “sweet spot similarity factory” and changed the tf-idf values.
- Installed a plugin for synonym expansion at query time,which gives more weightage to original word than synonyms for better relevancy.
- Tried to remove the near duplicates terms (using facet parameters in query) from the search results, as some of the tweets have almost same content with different id’s.

D. Results Summary

IR Model	Original MAP	Final MAP
Divergence From Randomness (DFR) Model	0.6468	0.6869
Okapi BM25	0.6554	0.6913
Vector Space Model (VSM)	0.6469	0.6834



Comparison of IR Models before enhancement



Comparison of IR Models after enhancement