Evaluation of IR Models

SRI SAI BHARGAV KORITALA
Department of Computer Science
University at Buffalo
Buffalo NY, 14214
srisaibh@buffalo.edu

Abstract

This project is aimed at implementing and evaluating different IR models. Observe the performance of your IR system using the results of TREC eval and improve the search results by adjusting the various parameters accepted by the IR model using query parser, boosting methods and several other methods.

1 Introduction

Twitter information is used in three languages English, German and Russian. The twitter data were indexed to three different cores corresponding to various IR models using Solr. Three following I R models have been implemented:

1. Language Model
2. BM25
3. Divergence and Randomness Model (DFR)

For each template, the results for different queries are collected and evaluated using TREC eval rel ated judgements. TREC eval's output is used to equate models with the value of Mean Average Pre cision(MAP).

2 Dataset

Twitter information was asked to ingest in to the Solr, which is stored in a JSON format. The data was collected in three different language that were included into the dataset as text_en for English, text_de for German and text_ru for Russian. A sample tweet is as follows:

{
"lang": "de",
"text_de": "RT @JulianRoepcke: ARTIKEL @BILD \nRussische Luftschläge in Syrien\nAssad",
"text_en": "",
"tweet_urls": ["http://www.bild.de/politik/ausland/syrien-krise/assad-isis-syrien42971016.bild.html"],
"text_ru": "",
"id": 653278482517110785,
"tweet_hashtags": []
}

3 Definitions

3.1 Language Model

All similarity frameworks smoothen scores based on unseen words(i.e. document length). We have used Dirichlet language model, this model does Bayesian smoothing using Dirichlet priors which implies how the actual term frequency stack up to what a Dirichlet distribution would assume to be "normal". There is a configurable parameter (mu) which controls smoothing, if the value of mu is high the scores will not change abruptly.

```
<similarity class="solr.LMDirichletSimilarityFactory">
      <float name="mu">6</float>
</similarity>
```

3.2 BM25

BM25 is a type of probabilistic IR model and default model in solr, which is an upgrade of TF-IDF. There are two configurable terms: k1 and b. A higher k1 implies higher ceiling, but it also makes document length normalization more dynamic(i.e. longer documents will be penalized more). In this model, the length doesn't get multiplied to the score directly. Instead, we get ratio between document length and average length of all documents in the index. Parameter b controls length normalization : higher b makes length matter more.

```
<similarity class="solr.BM25SimilarityFactory">
  <float name="b">0.8</float>
  <float name="k1">2.0</float>
</similarity>
```

3.3 Divergence and Randomness Model(DFR)

DFR is a framework which includes multiple models and normalization techniques which share the same principle : term may occur in a document randomly, following a certain distribution. More the document diverges from configured random distribution, higher the score. Three components of DFR are as follows:

1. The base model, which defines random distribution.
2. An after-effect, which normalizes score of base model based on term-frequency.
3. The term frequency used by after-effect is normalized based on document length. In the given project we are asked to use

```
<similarity class="solr.DFRSimilarityFactory">
  <str name="basicModel">G</str>
  <str name="afterEffect">B</str>
  <str name="normalization">H2</str>
  <float name="c">1</float>

</similarity>
```

4 Improving IR Model

We have implemented several logics that have improved the model ranging from an increase of zero to the maximum value that we have mentioned in the results. Although each and every model has responded differently to several methods. We shall discuss in detail for each and every model individually

- We have implemented initially without and query parsing and obtained a MAP Score of 0.3453 for the BM25 and 0.22 for the LM and 0.352 for DFR models respectively.
- Additional query parsing by dismax has showed light improvements bringing up the MAP score to 0.45 for BM25, 0.41 to LM and 0.48 for DFR model respectively.
- Discotinued the dismax query and implemented the edismax or extended dismax has helped search among multiple fields of the json data with different boosts based on the word occurance and the rarity.
- Boosting has been done by using Query boosters by tweaking various parameters including qf which boosts specific fields.
- Boosting is also done based on the language for a given query and de-boosting the search response of the other languages than the detected language of the query.
- We also added commonly added synonym words into the synonyms.txt manually like interference, mediation, discarded, released, alien, emigrant, exile, evacuee, foreigner, advanced, aided, assure, establish, insure, protect, secure, provide, ended, established, resolved, appointment, authorization, charge, commissioning, correlative, correspondent, counterpart, parallel, bomber, radical, rebel, rebel, thug, adversary, aspirant, enemy, foe, rival.
- We have also modified the stopwords.txt to remove extra words while indexing to exclude words like rt, http, https.

4.1 Language Model

- Language model is found sensitive to synonyms and stopwords, as removal of the stopwords for language has increased the model efficiency.

MU value	MAP Score
6	0.6463
10	0.6451
2000	0.6433
3000	0.6414

Fig Change of MAP for different mu values.

- In the below image we find that at a mu value of 6 we have achieved a MAP value of 0.6463.

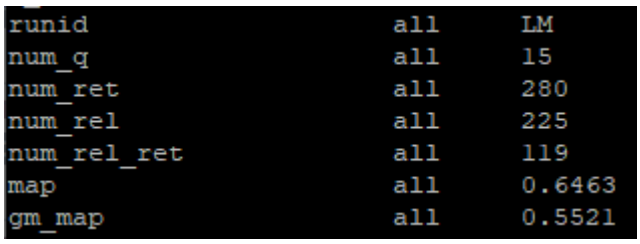

Fig. ScreenShot of Trec Eval for Language Model

- We have implemented the logic to boost the score and ranking of a query y using a langdetect library from python and upon detecting a language from the query we have boosted the corresponding text field for that language and parallel we have de boosted the text fields for other language or text fields

4.2 **BM25 Model**

- BM25 model is tweaked among various other parameters including the k1 and b values.

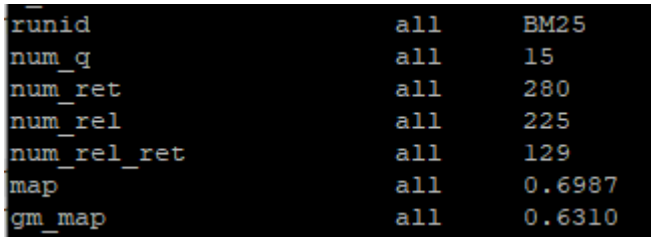

Fig. Screenshot of Trec Eval for BM25 Model

K1	**B**	**MAP Score**
1.2 (default)	0.75 (default)	0.6105
1.5	0.5	0.6292
1.6	0.6	0.6457
1.5	0.7	0.6870
1.8	0.9	0.6321
2.0	0.8	0.6987

Fig. Change in values of MAP score for corresponding k1 and b

4.3 **DFR Model**

- DFR is subjected to a basic model of G and Bernoulli's after affects i.e., 1st Normalization and a second normalization of H2.

C value	**MAP Score**
1	0.7055
5	0.6724
7	0.6954

Fig. Change in MAP value for corresponding c values in DFR similarity

- We have manually modified the value of c and have calculated the MAP score.

```
runid                   all     DFR
num_q                   all     15
num_ret                 all     280
num_rel                 all     225
num_rel_ret             all     130
map                     all     0.7055
gm_map                  all     0.6382
```

Fig MAP Score for DFR from TREC_eval

5 Results

We have gained the maximum MAP score for DFR model with a MAP value of 0.7055.

IR Model	MAP score
DFR	0.7055
BM25	0.6987
LM	0.6463

i. We can observe that for the given ingestion of tweets DFR model has a better performance on the data as compared to BM25 and LM.

ii. The difference in MAP is affected by the no of relevant docs that are retrieved from the query. Although the DFR has a better performance the difference in retrieved documents is very less between DFR and BM25 while there is a significant difference in in LM and the rest of the IR models that have been implemented.

iii. Tweaking of query by using synonyms and stopwords and boosting certain important terms based on certain usages like language and the context of the sentence has resulted in a significant increase in the MAP while that of the changes that have been implemented in the indexing part of the data during which we have increased the overall performance by a amount less than that of the query, but has a overall impact while taken at a whole document level.

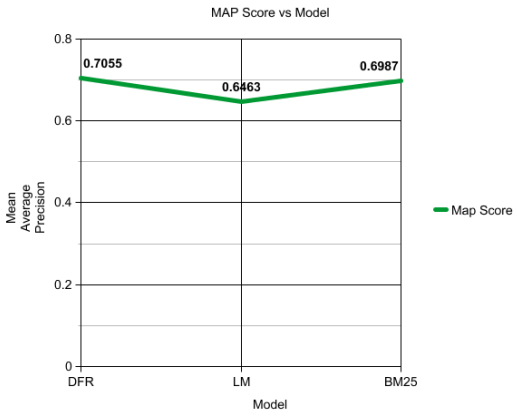

Fig. Map Score vs Model

6 Conclusion

Thus from the various analysis it is evident that the solr search engine gives desirable results when we use DFR model with the following parameters for model and edismax query parser.

MAP(DFR) > MAP(BM25) >>> MAP(LM)

References

1. http://lucene.apache.org/solr/7_0_0/solr-core/org/apache/solr/search/similarities/package-summary.html
2. http://wiki.apache.org/solr/SolrRelevancyFAQ#How_can_I_make_exact-case_matches_score_higher
3. https://cwiki.apache.org/confluence/display/solr/The+Standard+Query+Parser
4. http://trec.nist.gov/trec_eval/
5. https://cwiki.apache.org/confluence/display/solr/Common+Query+Parameters
6. http://lucene.apache.org/solr/7_0_0/solr-core/org/apache/solr/search/similarities/package-summary.html
7. http://lucene.apache.org/solr/guide/7_5/other-schema-elements.html#OtherSchemaElements-Similarity
8. https://lucene.apache.org/solr/8_1_0/solr-core/org/apache/solr/search/similarities/LMDirichletSimilarityFactory.html