
Evaluation of IR Models

Prudhveer Reddy Kankar
50320121

Department of Computer Science
University at Buffalo
Buffalo NY, 14214
prudhvee@buffalo.edu

Abstract

The purpose of this project is to evaluate the performance of various IR models. We use TREC_eval to evaluate the performance and try to improve the results by tweaking various parameters and by using query parsers and various boosting techniques.

1 Introduction

In this project take input of twitter data of three languages English, German and Russian. This data is being indexed in Solr and we then apply three different IR models :

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

Then for each model we evaluate the performance using TREC_eval for the given queries. In later stage we use MAP values to compare the performance of all the three IR models implemented.

2 Dataset

The dataset used is a Twitter data which is in a JSON format. There are three languages in the dataset and we have been given 3440 tweets to be implemented.

The sample of a tweet is given as follows :

```
{  "lang": ,
  "id": ,
  "text_de": ,
  "text_en": ,
  "text_ru": ,
  "tweet_urls": [ ],
  "tweet_hashtags": []
}
```

3 Definitions

3.1 Language Model

Language model attempts to model query generation process. Documents are ranked by the probability that a query would be observed as a random sample from the respective document model. We use the below code in the managed-schema to apply the language model for indexing.

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

3.2 BM25

BM25 is one of the probabilistic model and is used as a default model in Solr. There are hyper parameters in BM25 k_1 and b which are configured and changed based on requirements. Higher k_1 means that there is higher ceiling and the longer documents are penalized more. Parameter b controls the length normalization and makes the length value more significant. We use the below code in the managed-schema to apply the BM25 model for indexing.

```
<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 also a type of probabilistic model. The DFR is based on the idea that the more the divergence of the within-document term-frequency from its frequency within the collection, the more the information carried by the word t in document d . We use the below code in the managed-schema to apply the BM25 model for indexing. We use the below code in the managed-schema to apply the BM25 model for indexing.

```
<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 three models and applied various modifications and filters.

- We have used dismax query parser for searching among the various fields of JSON data.
- Boosting has been done by using Query boosters by tweaking various parameters including qf which boosts specific fields.
- Boosting is also done for the query based on the language of the query.
- The stop words are also modified to remove few extra words.
- Few extra synonym words are also added into the synonyms.txt.

4.1 Language Model

The language model is found to be reacting more to the changes in synonyms and the stop words. We have found significant increase in the efficiency of the model by removing them.

MU Value	Map Score
----------	-----------

6	0.6463
2000	0.6451
3000	0.6433

4.2 BM25

The efficiency of BM25 model is checked by tweaking various parameters including k_1 and b values.

K1	B	Map Score
1.5	0.5	0.6292
1.6	0.6	0.6457
1.5	0.7	0.6870
2.0	0.8	0.6987

4.3 DFR

DFR is subjected to a basic model of G and Bernoulli's after affects i.e., 1st Normalization and a second normalization of H_2 . The value of c has been manually changed and calculated the MAP scores which are found in the below table.

C Value	MAP Score
1	0.7055
5	0.6724

5 Results

The below are the Trec_eval scores for various models implemented.

```
runid      all      BM25
num_q      all      15
num_ret    all      280
num_rel    all      225
num_rel_ret all      129
map         all      0.6987
gm_map      all      0.6310
Rprec       all      0.6694
bpref       all      0.7011
recip_rank  all      1.0000
iprec_at_recall_0.00 all 1.0000
iprec_at_recall_0.10 all 0.9644
iprec_at_recall_0.20 all 0.9286
iprec_at_recall_0.30 all 0.8863
iprec_at_recall_0.40 all 0.8635
iprec_at_recall_0.50 all 0.8203
iprec_at_recall_0.60 all 0.6516
iprec_at_recall_0.70 all 0.5710
iprec_at_recall_0.80 all 0.4385
iprec_at_recall_0.90 all 0.3279
iprec_at_recall_1.00 all 0.3279
P_5         all      0.8400
P_10        all      0.6667
P_15        all      0.5289
P_20        all      0.4300
P_30        all      0.2867
P_100       all      0.0860
P_200       all      0.0430
P_500       all      0.0172
P_1000      all      0.0086
```

Trec_eval score for BM25

```
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
Rprec       all      0.6890
bpref       all      0.7124
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 0.9009
iprec_at_recall_0.40 all 0.8742
iprec_at_recall_0.50 all 0.8315
iprec_at_recall_0.60 all 0.6685
iprec_at_recall_0.70 all 0.5654
iprec_at_recall_0.80 all 0.4459
iprec_at_recall_0.90 all 0.3353
iprec_at_recall_1.00 all 0.3353
P_5         all      0.8400
P_10        all      0.6733
P_15        all      0.5289
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-172-31-46-159:~/trec_eval-9.0.7$
```

Trec_eval score for DF

```
runid      all      LM
num_q      all      15
num_ret    all      280
num_rel    all      225
num_rel_ret all      115
map         all      0.6433
gm_map      all      0.5635
Rprec       all      0.6243
bpref       all      0.6502
recip_rank  all      1.0000
iprec_at_recall_0.00 all 1.0000
iprec_at_recall_0.10 all 0.9762
iprec_at_recall_0.20 all 0.8900
iprec_at_recall_0.30 all 0.8524
iprec_at_recall_0.40 all 0.8440
iprec_at_recall_0.50 all 0.6808
iprec_at_recall_0.60 all 0.4782
iprec_at_recall_0.70 all 0.4304
iprec_at_recall_0.80 all 0.3892
iprec_at_recall_0.90 all 0.3000
iprec_at_recall_1.00 all 0.3000
P_5         all      0.8267
P_10        all      0.6200
P_15        all      0.4711
P_20        all      0.3833
P_30        all      0.2556
P_100       all      0.0767
P_200       all      0.0383
P_500       all      0.0153
P_1000      all      0.0077
ubuntu@ip-172-31-46-159:~/trec_eval-9.0.7$
```

Trec_eval score for LM

6 Conclusion

Model	MAP SCORE
BM25 Model	0.6987
DFR Model	0.7055
Language Model	0.6433

The maximum score was observed for DFR model which is 0 . 7055 .