Project Part B: Evaluation of IR models

Team Name: CAIR

**Contributors:**

Akash Mandole (50168447)

Chintan Thakker (50167864)

**INDEX**

**INTRODUCTION**

The project aims to understand and implement various IR models. We will be using various evaluation measures to evaluate the IR system and improve the search result based on the understanding of evaluation. The data used consists of tweets from three different languages English, German and Russian. The topic covered is mostly “European refugee crisis”.

Following IR models are implemented:

1. Vector Space Model
2. BM25 Model
3. Language Model

Following evaluation measures are considered:

1. nDCG
2. MAP
3. F0.5
4. BPREF

**DATA ANALYSED**

The data utilized for this project is twitter data collected in json format on topic European refugee crisis in English, German and Russian. We will be using **train.json.** The file contains tweets in the following format:

{

"lang": ,

"id": ,

"text\_de": ,

"text\_en": ,

"text\_ru": ,

"created\_at": ,

"tweet\_urls": [ ],

"tweet\_hashtags": []

}

**IR MODELS USED**

Following IR models are used:

1. **Vector Space Model:** Solr by default uses **DefaultSimilarityFactory.** Which is a factory for **DefaultSimilarity.** It is based on default scoring implementation, based upon the Vector Space Model. Solr combines Boolean model (BM) of Information Retrieval with Vector Space Model (VSM) of Information Retrieval - documents "approved" by BM are scored by VSM.In VSM, documents and queries are represented as weighted vectors in a multi-dimensional space, where each distinct index term is a dimension, and weights are Tf-idf values. VSM score of document d for query q is the Cosine Similarity of the weighted query vectors V(q) and V(d):

cosine-similarity (q, d) = V(q) · V(d) / |V(q)| |V(d)|

Solr refines VSM score for both search quality and usability as follows:

score(q,d) = coord-factor(q,d) · query-boost(q) · V(q) · V(d) / |V(q)| · doc-len norm(d) · doc-boost(d)

1. **BM25:** Solr uses **BM25SimilarityFactory** to implement this, which in turn is a factory for **BM25Similarity.** Introduced in Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at TREC-3. In Proceedings of the Third Text Retrieval Conference (TREC 1994). Gaithersburg, USA, November 1994. BM25 is a bag-of-words retrieval function that ranks a set of documents based on the query terms appearing in each document, regardless of the inter-relationship between the query terms within a document. Given a query Q, containing keywords q1,…qn, the BM25 score of a document D is:

 \text{score}(D,Q) = \sum_{i=1}^{n} \text{IDF}(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{\text{avgdl}})},

where f (q1, D) is qi’s term frequency in the document D, |D| is the length of the document D in words, and avgdl is the average document length in the text collection from which documents are drawn. k1 and b are free parameter IDF (qi) is the IDF (inverse document frequency) weight of the query term qi.

1. **LM:** Solr uses various similarities like LMDirichletSimilarity, LMJelinekMercerSimilarity, etc. to implement Language models. We will be using **LMJelinekMercerSimilarityFactory** which implements LMJelinekMercerSimilarity. Language model based on the Jelinek-Mercer smoothing method. From Chengxiang Zhai and John Lafferty. 2001. A study of smoothing methods for language models applied to Ad Hoc information retrieval. In Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '01). ACM, New York, NY, USA, 334-342. The model has a single parameter, λ. According to the paper, the optimal value depends on both the collection and the query. The optimal value is around 0.1 for title queries and 0.7 for long queries.

**IMPROVEMENTS ON IR SYSTEM**

This section describes various techniques used to improve the IR system. It includes examining the query provided, advanced query processing like boosting the query based on various fields. We will also improve the performance based on expanding the query, additional analyzer and tokenizer. We will tweak the parameters of the IR model to make it more suitable to the query. Filters used in the query processing will also be analysed. All improvements will be judged on measures like nDCG MAP F0.5 BPREF.

**Tweaking IR model Parameters**

**BM25**: This similarity has the following options:

* k1: Controls non-linear term frequency normalization (saturation).
* B: Controls to what degree document length normalizes tf values.

Following table and graph illustrates the change in these parameters and the results observed over nDCG.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| nDCG |  |  |  |  |  |
|  |  |  |  |  |  |
| No Modification : | K1.0 : | K2.0 : | K1.0\_B0.5 | B0.5 | B0.9 |
|  |  |  |  |  |  |
| 0.7648 | 0.7648 | 0.6937 | 0.7647 | 0.7647 | 0.6937 |
| 0.8325 | 0.8327 | 0.8224 | 0.8396 | 0.8451 | 0.8224 |
| 0.9406 | 0.942 | 0.9189 | 0.9652 | 0.9652 | 0.9189 |
| 0.8699 | 0.8699 | 0.869 | 0.8698 | 0.8698 | 0.8691 |
| 0.7244 | 0.7244 | 0.7244 | 0.7244 | 0.7244 | 0.7244 |
| 0.7327 | 0.7537 | 0.7327 | 0.7537 | 0.7537 | 0.7327 |
| 0.9448 | 0.9448 | 0.9448 | 0.9448 | 0.9448 | 0.9448 |
| 1 | 1 | 1 | 1 | 1 | 1 |
| 0.9062 | 0.9062 | 0.9062 | 0.9062 | 0.9062 | 0.9062 |
| 0.9963 | 0.9963 | 0.9963 | 0.9963 | 0.9963 | 0.9963 |
| 0.9424 | 0.9424 | 0.9424 | 0.9439 | 0.9444 | 0.9424 |
| 0.7815 | 0.7815 | 0.7815 | 0.7828 | 0.7828 | 0.7815 |
| 0.703 | 0.703 | 0.6962 | 0.7037 | 0.7047 | 0.6967 |
| 0.7957 | 0.7973 | 0.7878 | 0.7997 | 0.8002 | 0.7878 |
| Series 1 | Series 2 | Series 3 | Series 4 | Series 5 | Series 6 |

Following table and graph illustrates the change in same parameters and the results observed over MAP.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MAP |  |  |  |  |
|  |  |  |  |  |
| No modification | K1.0 | K2.0 | b0.5 | b0.9 |
|  |  |  |  |  |
| 0.4188 | 0.4188 | 0.3258 | 0.4185 | 0.3258 |
| 0.7447 | 0.7456 | 0.7499 | 0.7655 | 0.7499 |
| 0.7962 | 0.8065 | 0.8017 | 0.8121 | 0.8018 |
| 0.5448 | 0.5448 | 0.5393 | 0.5443 | 0.5402 |
| 0.5 | 0.5 | 0.5 | 0.5 | 0.5 |
| 0.4837 | 0.5251 | 0.4837 | 0.5251 | 0.4837 |
| 0.75 | 0.75 | 0.75 | 0.75 | 0.75 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 0.9861 | 0.9861 | 0.9861 | 0.9861 | 0.9861 |
| 0.814 | 0.814 | 0.814 | 0.7954 | 0.814 |
| 0.2244 | 0.2244 | 0.2244 | 0.2267 | 0.2244 |
| 0.5424 | 0.5424 | 0.5138 | 0.5495 | 0.5175 |
| 0.6289 | 0.6327 | 0.6206 | 0.6338 | 0.6209 |
| Series 1 | Series 2 | Series 3 | Series 4 | Series 5 |

**Conclusion:** It is evident that the tweaking the parameter b to 0.5 and keeping K1 as default (1.2) yields the best results over all measures.

**LM:** LM Jelinek Mercer similarity has the following options:

* Lambda: The optimal value depends on both the collection and the query. The optimal value is around 0.1 for title queries and 0.7 for long queries. Default to 0.1.

Following table and graph illustrates the change in this parameter and the results observed over nDCG.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| nDCG |  |  |  |  |  |
|  |  |  |  |  |  |
| No Modification | lamda 0.1 | lamda 0.2 | lamda 0.3 | lamda 0.5 | lamda 0.7 |
|  |  |  |  |  |  |
| 0.7752 | 0.7766 | 0.7766 | 0.7766 | 0.7766 | 0.7752 |
| 0.8343 | 0.8306 | 0.8306 | 0.8303 | 0.8296 | 0.8343 |
| 0.9648 | 0.9631 | 0.9637 | 0.9637 | 0.9669 | 0.9648 |
| 0.868 | 0.868 | 0.868 | 0.868 | 0.868 | 0.868 |
| 0.7244 | 0.7244 | 0.7244 | 0.7244 | 0.7244 | 0.7244 |
| 0.7737 | 0.7737 | 0.7737 | 0.7737 | 0.7737 | 0.7737 |
| 0.9448 | 0.9448 | 0.9448 | 0.9448 | 0.9448 | 0.9448 |
| 1 | 1 | 1 | 1 | 1 | 1 |
| 0.9062 | 0.9062 | 0.9062 | 0.9062 | 0.9062 | 0.9062 |
| 0.9963 | 0.9963 | 0.9963 | 0.9963 | 0.9963 | 0.9963 |
| 0.9528 | 0.9478 | 0.9478 | 0.9478 | 0.9509 | 0.9528 |
| 0.7838 | 0.7842 | 0.7842 | 0.7842 | 0.7842 | 0.7838 |
| 0.887 | 0.887 | 0.887 | 0.887 | 0.887 | 0.887 |
| 0.8151 | 0.8145 | 0.8145 | 0.8145 | 0.8149 | 0.8151 |
| Series 1 | Series 2 | Series 3 | Series 4 | Series 5 | Series 6 |

Following table and graph illustrates the change in this parameter and the results observed over MAP

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MAP |  |  |  |  |  |
|  |  |  |  |  |  |
| No Modification | lamda 0.1 | lamda 0.2 | lamda 0.3 | lamda 0.5 | lamda 0.7 |
|  |  |  |  |  |  |
| 0.4133 | 0.4188 | 0.4188 | 0.4188 | 0.4186 | 0.4133 |
| 0.7027 | 0.7117 | 0.7117 | 0.7102 | 0.7063 | 0.7027 |
| 0.7726 | 0.7962 | 0.8013 | 0.8013 | 0.7866 | 0.7726 |
| 0.5365 | 0.5365 | 0.5365 | 0.5365 | 0.5365 | 0.5365 |
| 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 |
| 0.5661 | 0.5661 | 0.5661 | 0.5661 | 0.5661 | 0.5661 |
| 0.75 | 0.75 | 0.75 | 0.75 | 0.75 | 0.75 |
| 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 |
| 0.9861 | 0.9861 | 0.9861 | 0.9861 | 0.9861 | 0.9861 |
| 0.795 | 0.764 | 0.764 | 0.764 | 0.7817 | 0.795 |
| 0.2284 | 0.2292 | 0.2292 | 0.2292 | 0.2292 | 0.2284 |
| 0.6811 | 0.6811 | 0.6811 | 0.6811 | 0.6811 | 0.6811 |
| 0.638 | 0.6385 | 0.6389 | 0.6388 | 0.6387 | 0.638 |
| Series 1 | Series 2 | Series 3 | Series 4 | Series 5 | Series 6 |

**Conclusion:** Though tweaking the parameter lambda does not yield a significant improvement for all measures however a lambda value of 0.2 is a good trade of between MAP and nDCG value, hence ideal in this scenario.

The purpose of this report is to analyse query result, query texts, ground truth result and the TREC\_eval result, to gain an intuition of the performance of our IR system. We will be evaluating our system based on various measures, including F0.5, nDCG, MAP etc.

**Improving Vector Space Model:-**

**QUERY EXPANSION:-** Query expansion is the process of reformulating a seed query to improve retrieval performance in information retrieval operations. The goal of query expansion in this regard is by increasing recall, precision can potentially increase (rather than decrease as mathematically equated). We have expanded queries 001, 005 and 012 by translating each one of them to all the three Languages. Thus the recall increases, however there might be some decrease in values of precision. To avoid this we will do Field Boosting to make sure goal of Query expansion is met.

Following tables show the values Before and after Query Expansion:-

|  |  |  |
| --- | --- | --- |
|  | Retrieved Documents  (num\_ret) |  |
|  |  |  |
|  | No Modification | Query Expansion |
|  |  |  |
| Query 001 | 283 | 1034 |
| Query 005 | 340 | 934 |
| Query 012 | 678 | 1395 |

|  |  |  |
| --- | --- | --- |
|  | MAP |  |
|  |  |  |
|  | No Modification | Query Expansion |
|  |  |  |
| Query 001 | 0.4188 | 0.3435 |
| Query 005 | 0.5000 | 0.6943 |
| Query 012 | 0.8140 | 0.5214 |

|  |  |  |
| --- | --- | --- |
|  | ndcg |  |
|  |  |  |
|  | No Modification | Query Expansion |
|  |  |  |
| Query 001 | 0.7648 | 0.7774 |
| Query 005 | 0.7244 | 0.7117 |
| Query 012 | 0.9424 | 0.8836 |

|  |  |  |
| --- | --- | --- |
|  | F\_0.5 |  |
|  |  |  |
|  | No Modification | Query Expansion |
|  |  |  |
| Query 001 | 0.0717 | 0.0297 |
| Query 005 | 0.0088 | 0.0064 |
| Query 012 | 0.0263 | 0.0534 |

|  |  |  |
| --- | --- | --- |
|  | bpref |  |
|  |  |  |
|  | No Modification | Query Expansion |
|  |  |  |
| Query 001 | 0.5575 | 0.7750 |
| Query 005 | 0.7027 | 0.7500 |
| Query 012 | 0.7726 | 0.6686 |

**Boosting the Query based on Different Fields:-** Due to Query expansion the increase in recall affects precision which we do not want. Thus we use Boosting of different fields to achieve Precision. If the original query is in English then we use boosting the field in query in following way:- text\_en^3+text\_de^0.5+text\_ru^2.

If the original query is in Russian then we use boosting the field in query in following way: - text\_en^3+text\_de^0.2+text\_ru^4

If the original query is in German then we use boosting the field in query in following way: - text\_en^2+text\_de^8+text\_ru^0.3

|  |  |  |
| --- | --- | --- |
|  | MAP |  |
|  |  |  |
|  | No Modification | Field Boosting |
|  |  |  |
| Query 001 | 0.4188 | 0.4450 |
| Query 005 | 0.5000 | 0.6553 |
| Query 012 | 0.8140 | 0.8165 |

|  |  |  |
| --- | --- | --- |
|  | ndcg |  |
|  |  |  |
|  | No Modification | Field Boosting |
|  |  |  |
| Query 001 | 0.7648 | 0.8346 |
| Query 005 | 0.7244 | 0.8781 |
| Query 012 | 0.9424 | 0.9536 |

|  |  |  |
| --- | --- | --- |
|  | F\_0.5 |  |
|  |  |  |
|  | No Modification | Field Boosting |
|  |  |  |
| Query 001 | 0.0717 | 0.0287 |
| Query 005 | 0.0088 | 0.0064 |
| Query 012 | 0.0263 | 0.0139 |

|  |  |  |
| --- | --- | --- |
|  | bpref |  |
|  |  |  |
|  | No Modification | Field Boosting |
|  |  |  |
| Query 001 | 0.5575 | 0.5800 |
| Query 005 | 0.7027 | 0.6875 |
| Query 012 | 0.7726 | 0.8343 |

**REFERENCES**

<http://lucene.apache.org/solr/4_0_0/solr-core/org/apache/solr/search/similarities/DefaultSimilarityFactory.html>

<http://lucene.apache.org/core/4_0_0/core/org/apache/lucene/search/similarities/BM25Similarity.html>

<https://www.elastic.co/guide/en/elasticsearch/reference/1.4/index-modules-similarity.html>

<http://lucene.apache.org/core/4_7_1/core/org/apache/lucene/search/similarities/LMJelinekMercerSimilarity.html>