Group 41

Lifestyle Forum Search Engine Implementation

SHABNAM KHODADADI- EC211233
DHRUV VERMA- EC211263

TUSHARA GOVINDA REDDY- EC211256
OM BHARATBHAI VAGHASIA- EC211242

Contents

- Problem Statement and Proposed Approach
- Dataset Definition
- Design Architecture
- Framework/Tools Used
- Implementation of Retrieval Models
- Evaluation
- Results
- Demo Video Link

- Note: Dear Evaluators, Due to the size of the recording we are providing the link to our SharePoint where the video has been hosted.

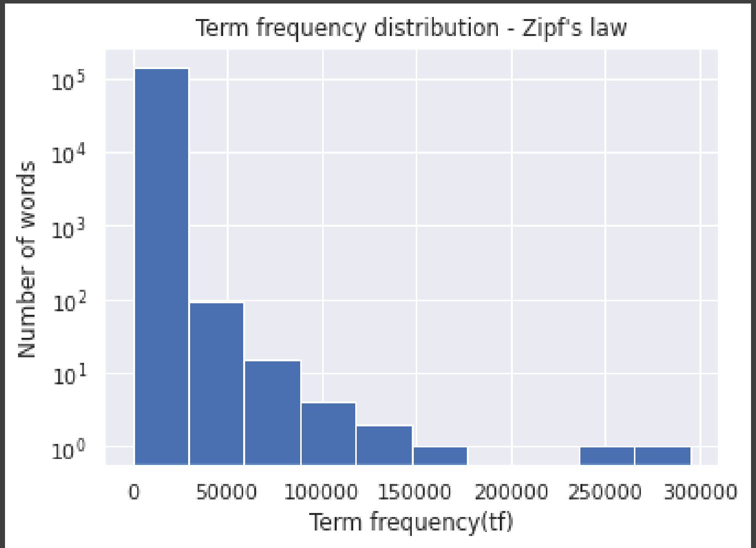
1. Problem Statement and Proposed Approach

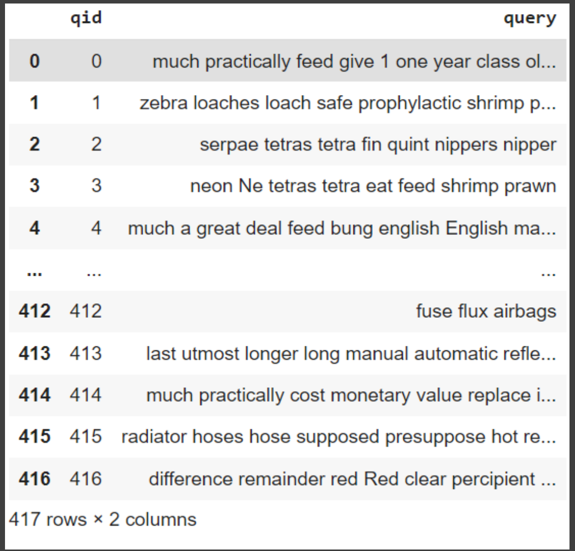
- Lifestyle forum are widely used by travel enthusiasts, marketers, lifestyle service/ product companies, complementary services etc. In order to find relevant results for a given lifestyle related query such as “best restaurant near me“ or “best cutlery store near me” we need real-time and semantic search engines to interpret the user location, demographic, preference and then provide relevant results.
- In our project we have implemented a Lifestyle Forum Search Engine which performs the following:
  - Retrieving semantically relevant items that don’t necessarily match the query terms exactly
  - Retrieving user-personalized items for the same search query
- Methodology :We propose to develop a prototype of lifestyle forum search engine which employs Bag of Word concepts for indexing, BM25, DFR-BM25, retrieval model, and TFIDF,TF for retrieval ranking along with Precision-R evaluation metrics to provide relevant query results.

2. Dataset Definition

- Data Description: Our data is the search results are from lifestyle-focused forums, including bicycles, coffee, crafts, diy, gardening, lifehacks, mechanics, music, outdoors, parenting, pets, sports, and travel. There are 268893 documents in our dataset

- Data distribution





	qid	query
0	0	much practically feed give 1 one year class ol...
1	1	zebra loaches loach safe prophylactic shrimp p...
2	2	serpae tetras tetra fin quint nippers nipper
3	3	neon Ne tetras tetra eat feed shrimp prawn
4	4	much a great deal feed bung english English ma...
...	...	...
412	412	fuse flux airbags
413	413	last utmost longer long manual automatic refle...
414	414	much practically cost monetary value replace i...
415	415	radiator hoses hose supposed presuppose hot re...
416	416	difference remainder red Red clear percipient ...

417 rows × 2 columns

Data Sample Text:

GenericDoc(doc_id='0', text="In my experience rabbits are very easy to housebreak. "
"They like to pee and poop in the same place every time, "
"so in most cases all you have to do is put a little bit of their waste in the litter box "
"and they will happily use the litter box. "
"It is very important that if they go somewhere else, "
"miss the edge or kick waste out of the box that you clean it up well "
"and immediately as otherwise those spots will become existing places to pee and poop. "

3. Design Architecture

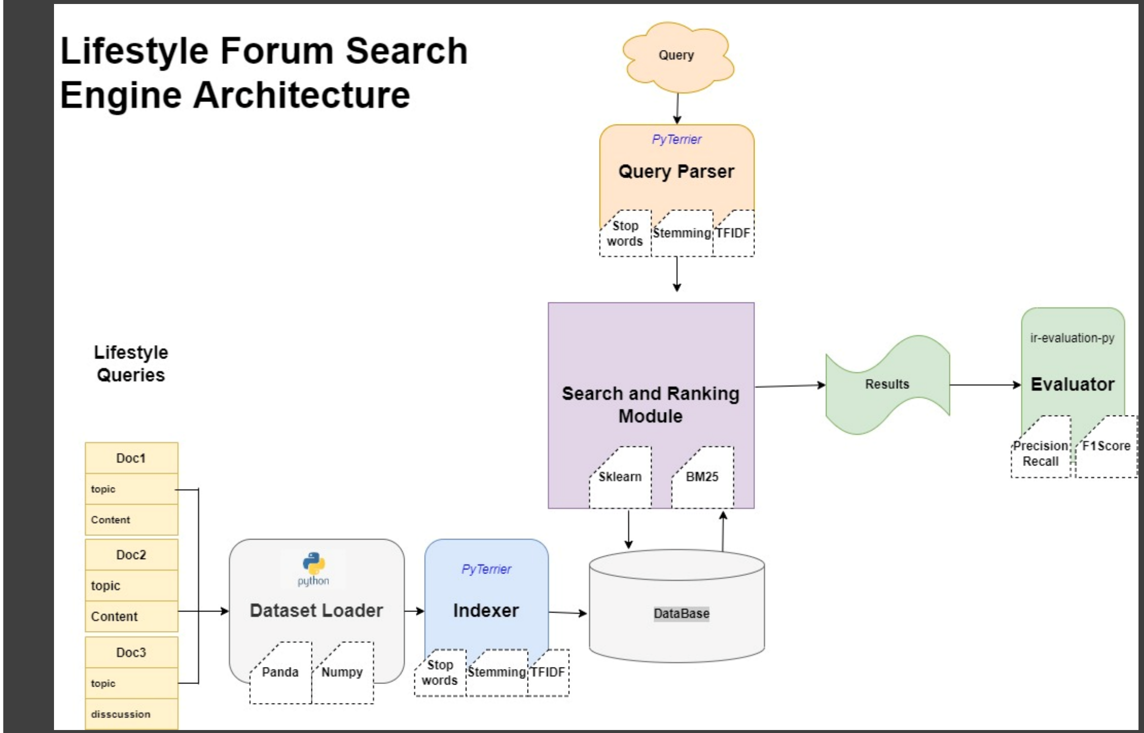
a. User Query Pre-processor is responsible for query parsing to be passed to the retrieval framework.

b. Dataset Loader will process the lifestyle dataset into workable format by performing cleaning, filtering, lowercasing. The indexer will implement specific indexing models on the dataset which includes:

- i. Stemming to reduce words to base words and remove suffixes
- ii. Removal of Stop Words, Punctuations, and special characters.
- iii. Indexing the words using Term Frequency and Inverse Document Frequency method

c. Search and Ranking Framework provides on API for retrieval model BM25, DFR-BM25 to be used on the parsed query. It will use index for scanning, ranking and retrieval of relevant recommendation based on the query passed. We propose to use the Best Match Okapi Model- BM25, DFR-BM25 as part of our framework. Retrieval Ranking, we are making use of TFIDF, TF.

d. Evaluator is responsible for the result evaluation based on relevant evaluation metrics such as Precision-R=5,10,15, NDCG, Relevant Ranking etc.



4. Framework/Tools Used

Framework	Description
Python (JupyterNotebook)	Application development
Pandas, NumPy	Dataset manipulation, pre-processing and structuring.
SKLearn- Feature extraction	Need to be able to convert the content of each string into vectors
PyTerrier	Indexing, Retrieval and Evaluation, Pipeline and Ranking
ir-evaluation-py	Effectiveness Evaluation Library for Python

5. Implementation of Retrieval Models

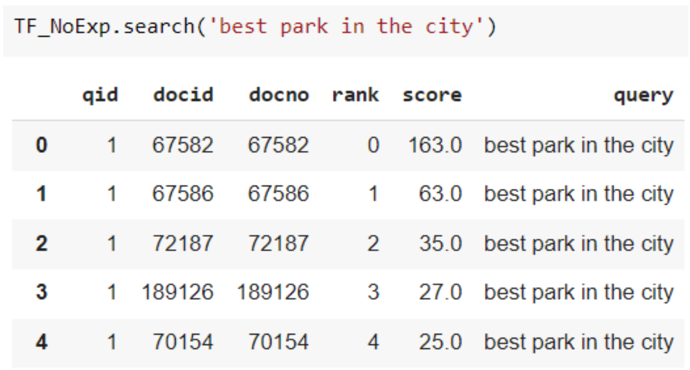
We implement four models:

- BM25
- DFR-BM25
- TFIDF
- TF

```
BM_25 = pt.BatchRetrieve(index_ref, wmodel='BM25')
DFR_BM25_NoExp = pt.BatchRetrieve(index_ref, wmodel='DFR_BM25')
TF_IDF_NoEXP = pt.BatchRetrieve(index_ref, wmodel='TF_IDF')
TF_NoExp = pt.BatchRetrieve(index_ref, wmodel='Tf')
```

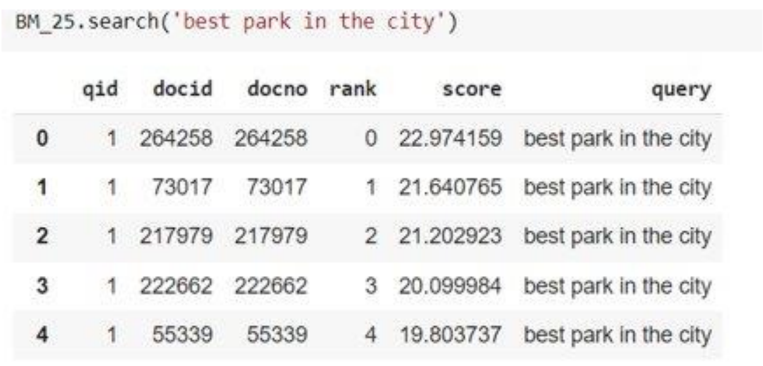
```
Relevant_document = pt.Experiment(
[BM_25, DFR_BM25_NoExp, TF_IDF_NoEXP, TF_NoExp],
topics,
qrels,
eval_metrics=["num_rel_ret"],
names=['BM25_NoExp', 'DFR_BM25_NoExp', 'TF_IDF_NoExp', 'TF_NoExp']
)
```

6. Evaluation



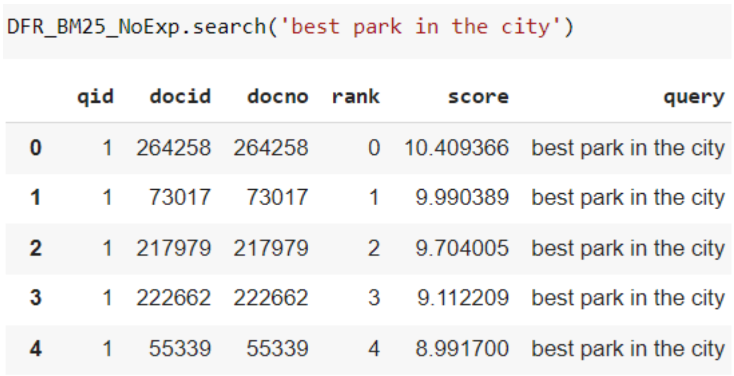
```
TF_NoExp.search('best park in the city')
```

	qid	docid	docno	rank	score	query
0	1	67582	67582	0	163.0	best park in the city
1	1	67586	67586	1	63.0	best park in the city
2	1	72187	72187	2	35.0	best park in the city
3	1	189126	189126	3	27.0	best park in the city
4	1	70154	70154	4	25.0	best park in the city



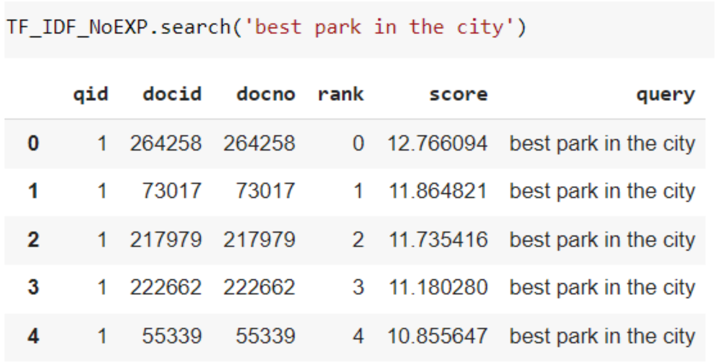
```
BM_25.search('best park in the city')
```

	qid	docid	docno	rank	score	query
0	1	264258	264258	0	22.974159	best park in the city
1	1	73017	73017	1	21.640765	best park in the city
2	1	217979	217979	2	21.202923	best park in the city
3	1	222662	222662	3	20.099984	best park in the city
4	1	55339	55339	4	19.803737	best park in the city



```
DFR_BM25_NoExp.search('best park in the city')
```

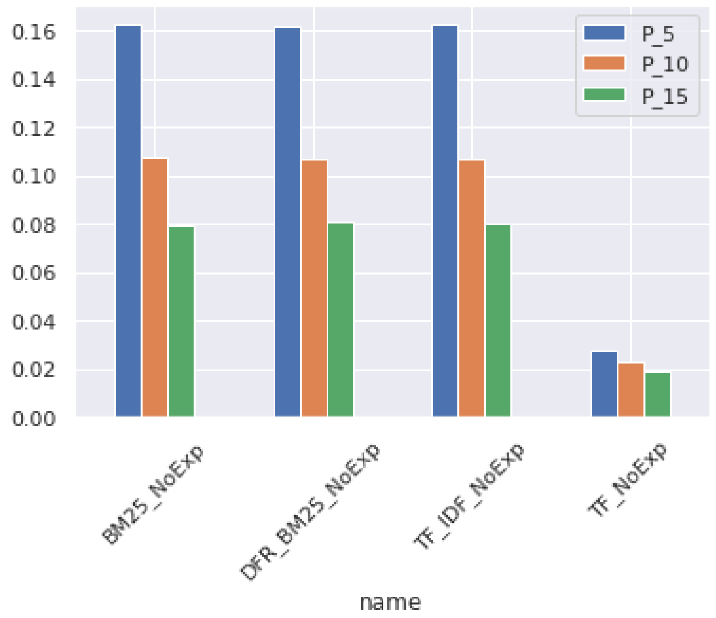
	qid	docid	docno	rank	score	query
0	1	264258	264258	0	10.409366	best park in the city
1	1	73017	73017	1	9.990389	best park in the city
2	1	217979	217979	2	9.704005	best park in the city
3	1	222662	222662	3	9.112209	best park in the city
4	1	55339	55339	4	8.991700	best park in the city



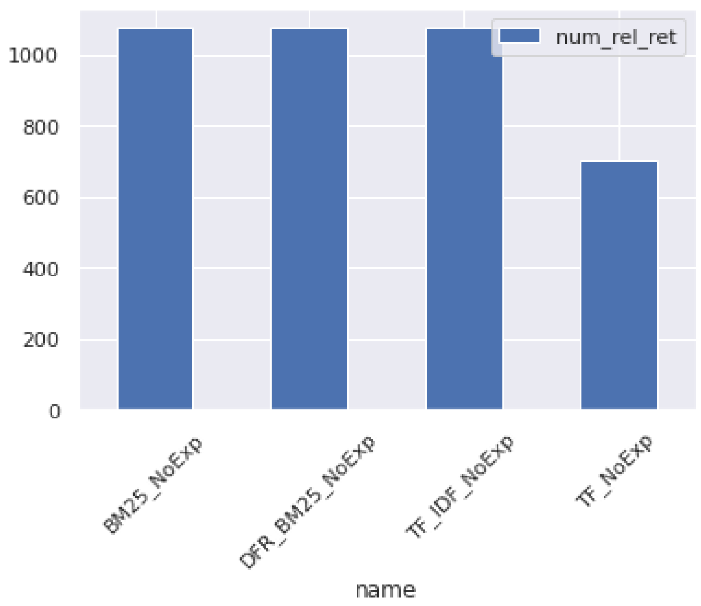
```
TF_IDF_NoEXP.search('best park in the city')
```

	qid	docid	docno	rank	score	query
0	1	264258	264258	0	12.766094	best park in the city
1	1	73017	73017	1	11.864821	best park in the city
2	1	217979	217979	2	11.735416	best park in the city
3	1	222662	222662	3	11.180280	best park in the city
4	1	55339	55339	4	10.855647	best park in the city

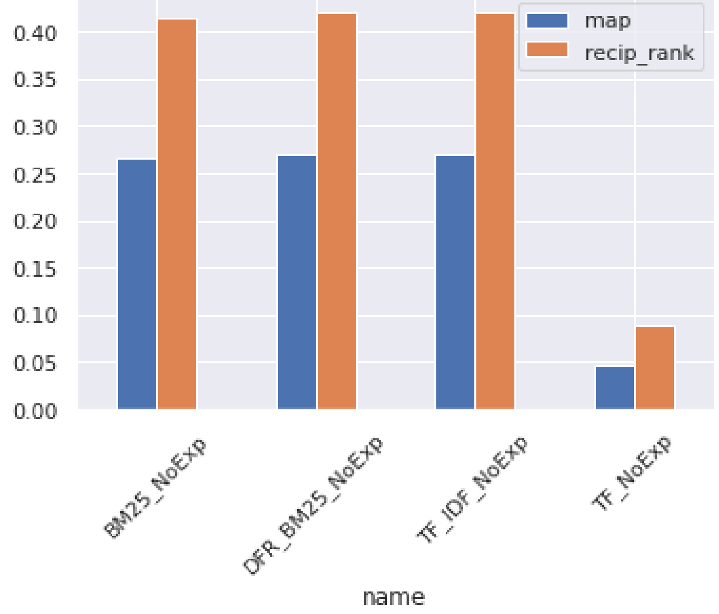
Precision @ R Scores



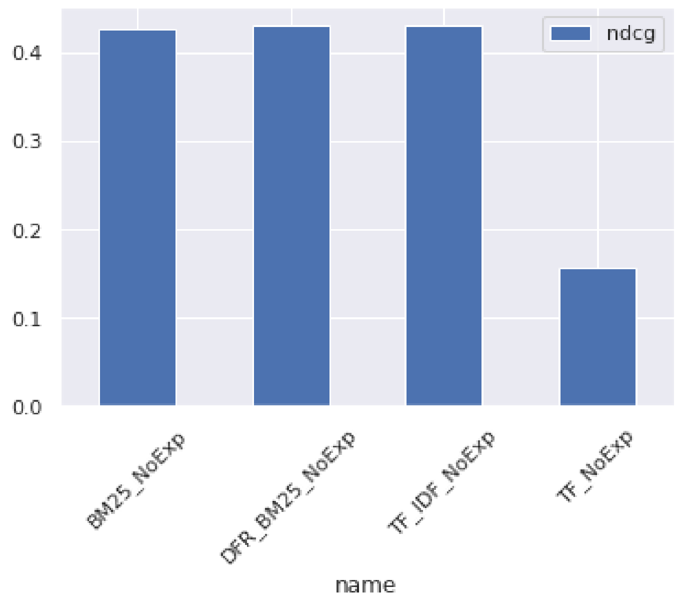
Relevant Documents Retrieved



Precision Rankings of Four Models



NDCG



7. Results

Query :

best park in the city

BM_25.search('best park in the city')

Demo Video Link on SharePoint

- Demonstratio Video_PG_41_IR_ASSGN_7.mp4
- https://qmulprod.sharepoint.com/:v:/r/sites/IRProject-Sem1/Shared%20Documents/General/PG_41_IR_Assignment3/Demonstratio%20Video_PG_41_IR_ASSGN_7.mp4?csf=1&web=1&e=yf11eR

- Note: Dear Evaluators, Due to the size of the recording we are providing the link to our SharePoint where the video has been hosted.

Thank you

Group 41_Information_Retrieval_Assignment3