MASTER IN DATA SCIENCE AND ICT INNOVATION

Information Extraction, Retrieval and Integration

Assignment 1: Building an Information Retrieval System

Yolanda de la Hoz Simón

Table of Contents

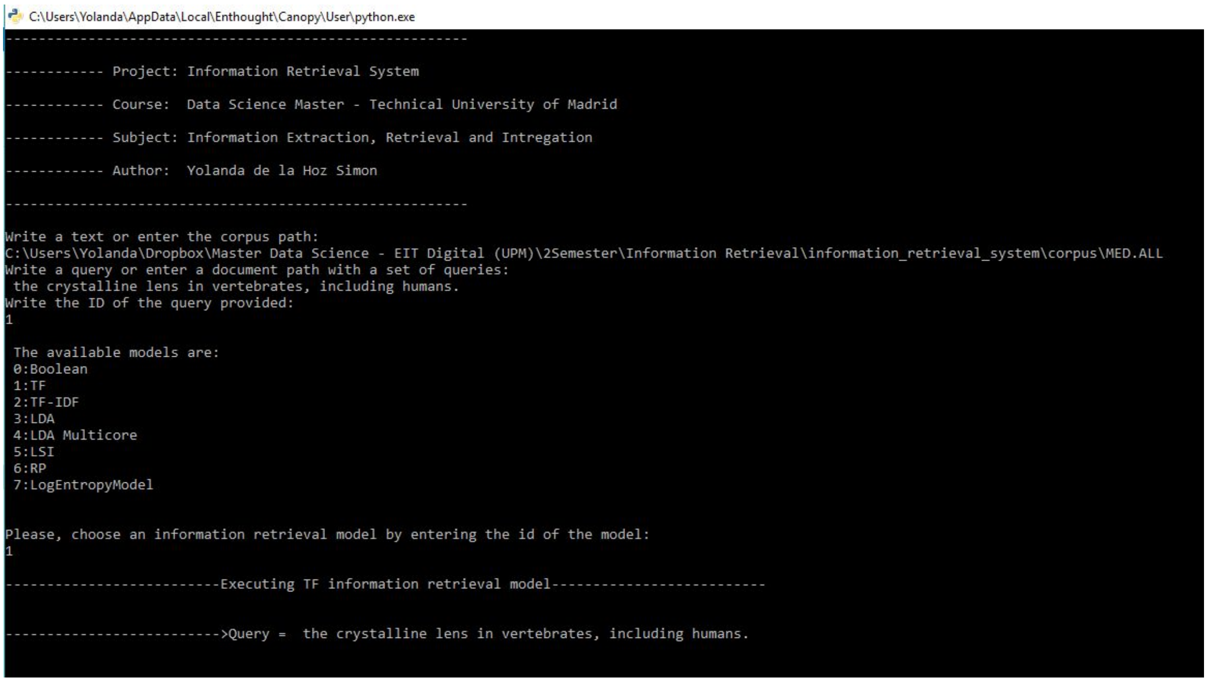
1. Introduction

The goal of this project is to implement an information retrieval system using Python, NLTK and GenSIM.

To build this system, it is provided a plain text MED.ALL that contains many documents related to life sciences. Each document is composed by 2 fields (.I and .W). The field .I contains a numeric ID that identifies the document, while the field .W contains the text of the document.

For this system is created 7 different versions of the IR using different weights for building the vectors representing documents and queries.

The program supplies an entry point to enable the user to launch queries and choose the desired IR system with the implemented models as it is shown in figure 1.



C:\Users\Yolanda\AppData\Local\Enthought\Canopy\User\python.exe

```
-----------------------------------------------------

----------- Project: Information Retrieval System

----------- Course:  Data Science Master - Technical University of Madrid

----------- Subject: Information Extraction, Retrieval and Intregation

----------- Author:  Yolanda de la Hoz Simon

-----------------------------------------------------

Write a text or enter the corpus path:
C:\Users\Yolanda\Dropbox\Master Data Science - EIT Digital (UPM)\2Semester\Information Retrieval\information_retrieval_system\corpus\MED.ALL
Write a query or enter a document path with a set of queries:
 the crystalline lens in vertebrates, including humans.
Write the ID of the query provided:
1

 The available models are:
 0:Boolean
 1:TF
 2:TF-IDF
 3:LDA
 4:LDA Multicore
 5:LSI
 6:RP
 7:LogEntropyModel


Please, choose an information retrieval model by entering the id of the model:
1

-------------------------Executing TF information retrieval model-------------------------


------------------------->Query =  the crystalline lens in vertebrates, including humans.
```

Figure 1: Information Retrieval Model Selection

The source code of this project as well as a brief description of the main components with an usage example is available in the following repository:

[1] https://github.com/yolanda93/information_retrieval_system

2. P/R Curves discussion

In the following figures are presented the P/R Curves generated by all the implemented IR. The P/R curves have been generated with the provided corpus and the query provided by the user. The query used to generate P/R curve shown in the figure 2 is:

" the crystalline lens in vertebrates, including humans."

For the relevance assessments and posterior precision and recall evaluation, it is used the provided file MED.REL with the real relevance documents and compared with the result obtained with each model after each retrieval, the individual precision has been interpolated to the scale (0-1) with increments of 0.1.

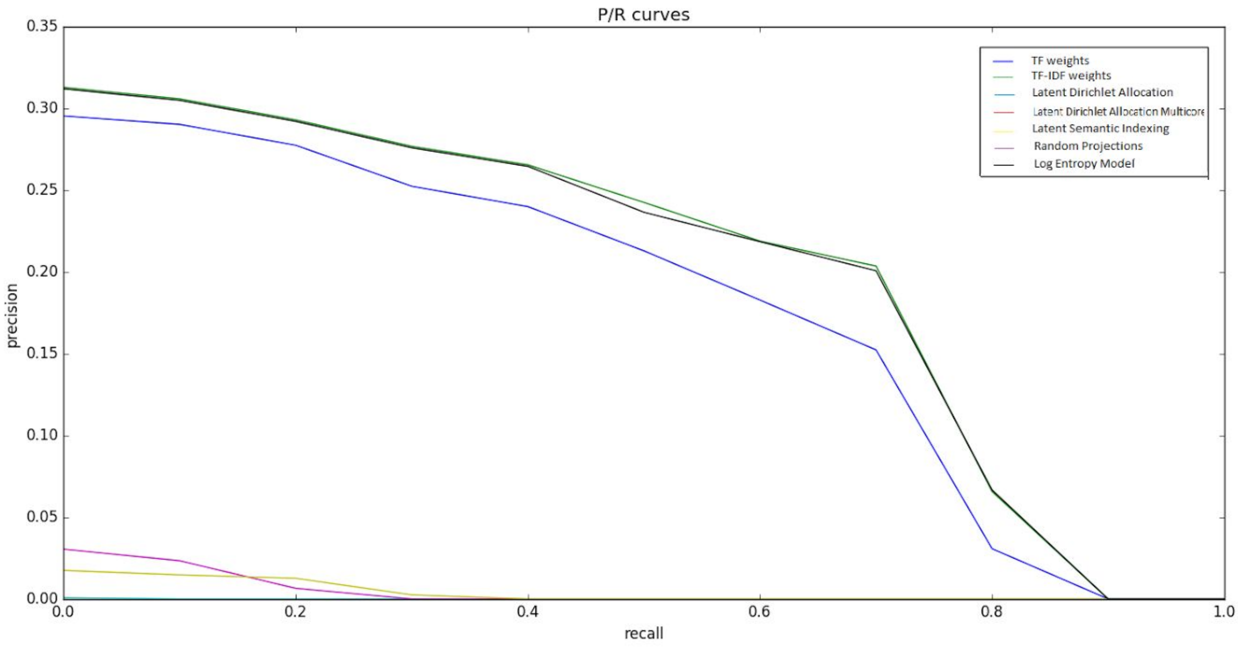


Figure 2: P/R curves plot - query 1

As it can be seen in the figure 2, the maximum recall achieved is 0.85 which means that not all the documents considered as relevant by the user are retrieved correctly by the IR. In addition, the P/R curves shows that TF, TF-IDF weights and Log Entropy models have very similar performance, while the others shows a really low performance.

The TF (inverse document frequency) assigns a lower value to words that appear in multiple documents (they are less significant), while the TF-IDF multiplies the frequency on the text with the IDF value.

Since the most efficient algorithm, the TF-IDF model, has an overall precision of 0.133928571429 and recall of 0.810810810811, it is considered to repeat the process with the second query provided in MED.QRY:

“the relationship of blood and cerebrospinal fluid oxygen concentrations
or partial pressures. a method of interest is polarography.”

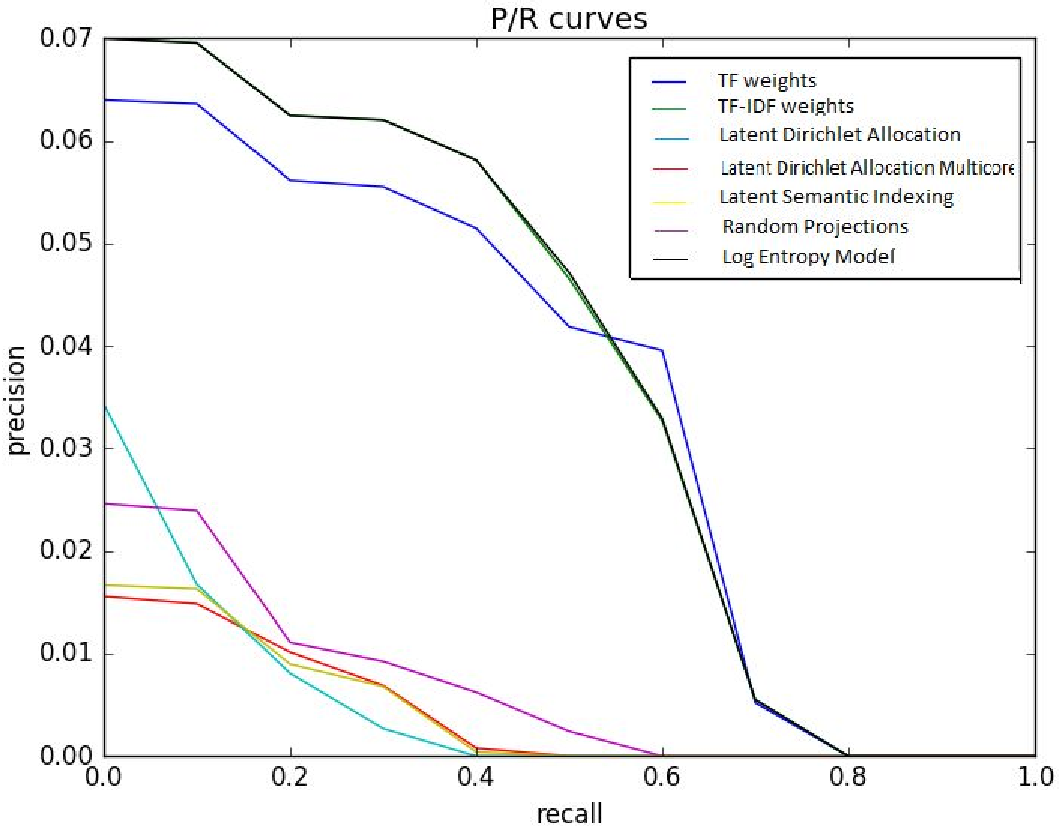


Figure 3: P/R curves plot - query 2

Nevertheless, the precision obtained also with this query is really low. Maybe it is due to the simplicity of query comparing the size of the corpus. In the following table it is gathered the overall precision and recall obtained with this query 2 and each model.

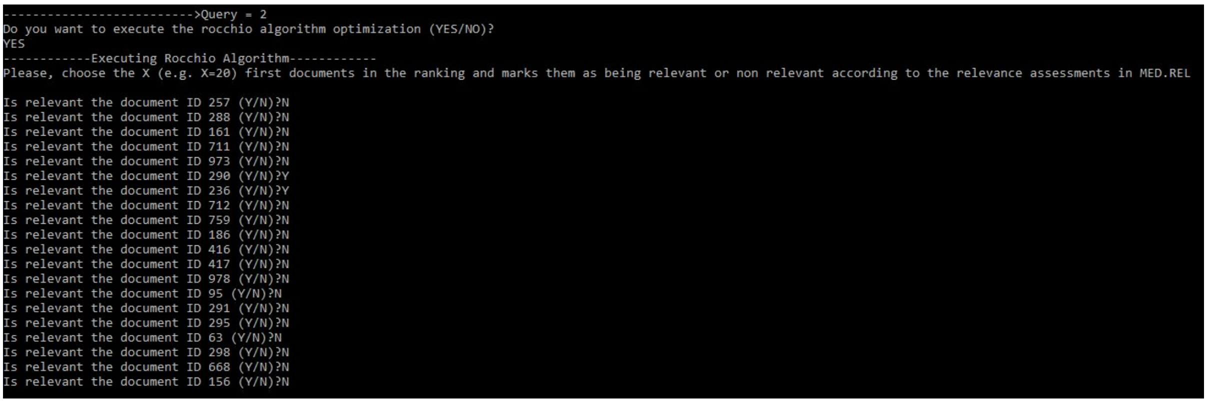
	TF-IDF	LDA	LDA (Multicore)	LSI	RP	Log Entropy
Precision	0.023	0.026	0.023	0.013	0.015	0.027
Recall	0.75	0.44	0.38	0.56	0.44	0.75

Figure 3: P/R curves plot - query 2

3. Rocchio Algorithm

In this phase it is implemented the rocchio algorithm that allows the user to improve the system's performance by incrementally reformulating the user query based on the relevance assessments provided by the user.

To implement this algorithm I followed the steps described in the moodle and explained in [1] which interaction with the user can be seen in the following figure. Once all the model are executed the IR implemented ask the user for the rocchio optimization, showing the top 20 ranked documents with the selected IR model and query, this can be seen in the figure 4.



```
------------------------->Query = 2
Do you want to execute the rocchio algorithm optimization (YES/NO)?
YES
------------Executing Rocchio Algorithm------------
Please, choose the X (e.g. X=20) first documents in the ranking and marks them as being relevant or non relevant according to the relevance assessments in MED.REL

Is relevant the document ID 257 (Y/N)?N
Is relevant the document ID 288 (Y/N)?N
Is relevant the document ID 161 (Y/N)?N
Is relevant the document ID 711 (Y/N)?N
Is relevant the document ID 973 (Y/N)?N
Is relevant the document ID 290 (Y/N)?Y
Is relevant the document ID 236 (Y/N)?Y
Is relevant the document ID 712 (Y/N)?N
Is relevant the document ID 759 (Y/N)?N
Is relevant the document ID 186 (Y/N)?N
Is relevant the document ID 416 (Y/N)?N
Is relevant the document ID 417 (Y/N)?N
Is relevant the document ID 978 (Y/N)?N
Is relevant the document ID 95 (Y/N)?N
Is relevant the document ID 291 (Y/N)?N
Is relevant the document ID 295 (Y/N)?N
Is relevant the document ID 63 (Y/N)?N
Is relevant the document ID 298 (Y/N)?N
Is relevant the document ID 668 (Y/N)?N
Is relevant the document ID 156 (Y/N)?N
```

Figure 4: Second query Rocchio Algorithm

Once the user has answered the relevant assessment questions according with the MED.REL text file for the provided query, in this case the query 2. It is executed the rocchio formula [figure 5] [2] to generate a new query.

$$\overrightarrow{Q_m} = \left(a \cdot \overrightarrow{Q_o}\right) + \left(b \cdot \frac{1}{|D_r|} \cdot \sum_{\overrightarrow{D_j} \in D_r} \overrightarrow{D_j}\right) - \left(c \cdot \frac{1}{|D_{nr}|} \cdot \sum_{\overrightarrow{D_k} \in D_{nr}} \overrightarrow{D_k}\right)$$

Figure 5: Rocchio formula

For simplicity the formula has been divided in three terms. First it is computed the term 2 and 3, which indicates the committed error (the weight sum difference between relevant and non-relevant documents) and then it is updated the weights of the original query generating the new modified query or vector Qm.

For this formula it is considered as parameters a, b and c, which I give the values (1, .75, .15), considering that following the examples executed before the system has really low recall, which means that not all relevant documents are retrieved by the system.

In addition, for the relevancy assessment, I considered as relevant document, the documents with has obtained more than 0.0 in the ranking and non relevant the rest.

Once is executed the formula, we have the following query:

> “the relationship of blood and cerebrospinal fluid oxygen concentrations or partial pressures. a method of interest is polarography.” + (coeffici appear)

In this formula it is introduced the terms “coeffici” and “appear” as they appear in the dictionary and according to the two words with maximum value in the updated weights. For this query it is executed again the TF-IDF IR model using the rocchio query optimization with the following results:

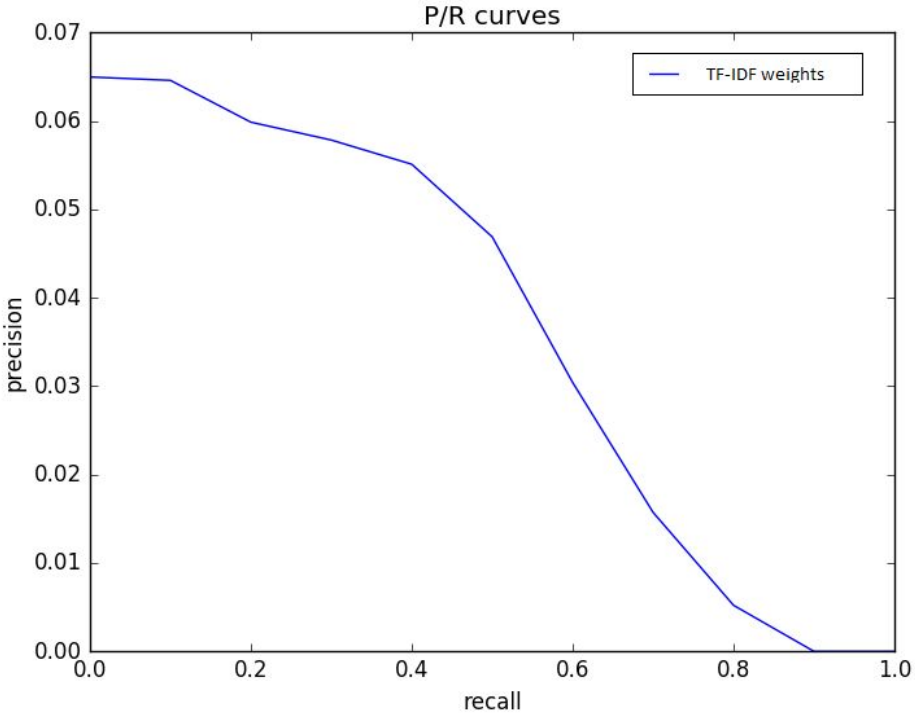


Figure 6: Improved Recall in TF-IDF (Second Query)

Although the precision is not improved, we get a higher recall 8.6 compared to the 8.1 obtained before.

References

[1] https://github.com/yolanda93/information_retrieval_system

[2] https://en.wikipedia.org/wiki/Rocchio_algorithm