Text Information Retrieval project.

Christian Raymond

10 janvier 2020

Table des matières

1 Introduction

The TP objective is to build a text search engine. This project is running over 3 sessions of 2 hours and will be evaluated. You will work by group of 4 students. Your implementation has to be evaluated on a provided dataset. A subset of 30 queries is provided in order to tune your system. The complete queries set will be provided only in the last session and your system will have to provide for each query a ranked list of relevant document. The performance of your system will be a part of the notation. You will also provide a small pdf document summarizing the conception of your engine.

2 Provided data

A compressed version of 3 files is provided under Moodle :

1. CISI.ALLnettoye is a collection of about 1460 books and papers descriptions ;
2. CISI.QRY is a set of 30 queries related to that collection ;
3. CISI.REL contains the relevance judgements for the documents *w.r.t.* the queries.

A tool that aims to evaluate your search engine performances

— eval.pl

Documents are separated from their neighbours in the collection with a special marker ; they all have a personal number, and are composed of a title and an abstract. A sample is given and explicited below.

```
.I 1
18 Editions of the Dewey Decimal Classifications
The present study is a history of the DEWEY Decimal
Classification. The first edition of the DDC was published
in 1876, the eighteenth edition in 1971, and future editions
will continue to appear as needed. In spite of the DDC's
long and healthy life, however, its full story has never
been told. There have been biographies of Dewey
that briefly describe his system, but this is the first
attempt to provide a detailed history of the work that
more than any other has spurred the growth of
librarianship in this country and abroad.
.I 2
Use Made of Technical Libraries
This report is an analysis of 6300 acts of use
in 104 technical libraries in the United Kingdom.
Library use is only one aspect of the wider pattern of
information use. Information transfer in libraries is
restricted to the use of documents. It takes no
account of documents used outside the library, still
less of information transferred orally from person
to person. The library acts as a channel in only a
proportion of the situations in which information is
transferred.
Taking technical information transfer as a whole,
there is no doubt that this proportion is not the
major one. There are users of technical information -
```

```
particularly in technology rather than science -
who visit libraries rarely if at all, relying on desk
collections of handbooks, current periodicals and personal
contact with their colleagues and with people in other
organizations. Even regular library users also receive
information in other ways.
```

— .I : indicates a new document ; the following number is the document identifier ;
— the title and the abstract compose the rest of the text and will be considered together.

The 3 following steps have to be done, using your favourite programming language.

3 Step 1 : collection indexing

For each document, identified by its .I field, the indexing of both title and abstract is produced through :

1. a tokenization : this treatment isolates words within the sequences of letters and symbols in the document. During this process, you choose what vocabulary you want to keep and do the associate processing (stemming, use of stopword list, *etc.*.) ;
2. a choice of indexing terms : elaborate a policy to choose the indexing terms among all the distinct words or stems (all the terms, only the most frequent ones, only the most discriminative ones, *etc.*). This choice mostly relies on a count of occurrences ;
3. define a weighting strategy for terms in the documents (*e.g.* TF.IDF) ;
4. the production of a representation vector for each document (a list of the weighted indexing terms). Note that this item has to be considered together with the following one ;
5. the production of inverted files for your system, for efficiency reasons. Thus an inverted file, corresponding to the structure indexing term $\rightarrow$ list of couples (document containing this term, weight), has to be produced from the above representation.

4 Step 2 : build the search engine

1. query indexing : to index the queries, use the same methodology and the same weighting schemes as for documents.
2. implementation of a search engine to answer the queries : a similarity measure between the indexing vectors of the queries and the documents has to be chosen and implemented. The expected output is a ranked list of documents calculated by the system in response to a given query.

5 Step 3 : evaluation

The performance of your IR system has to be evaluated, a script to do the evaluation is provided. This script output the next measures for each separated query as well as the overall result :

— Precision : the percentage of relevant documents among the ones provided by your system : ability to provide relevant documents ;
— Recall : percentage of relevant document provided by your system *w.r.t.* to the list expected in the database : ability to retrieve all the relevant documents ;
— F1 : a mix of the two previous measures ;
— Precision@1 : is the first ranked document relevant ? ;
— Precision@5 : percentage of relevant documents among the five first ranked documents.

6 Some NLP methods and tools (non exhaustive list)

To make your search engine more robust, you may use several pre-processing tools :

— a lot of segmenters or tokenizers are available ; so are stopword lists for several languages (*e.g.*, at `http://torvald.aksis.uib.no/corpora/1999-1/0042.html`, or Jean Véronis's list but several others exist) ;

— stemmers or morphological analyzers : Lovins's, Porter's (`http://www.tartarus.org/martin/PorterStemmer/`) or Paice-Huster's stemmers ; Flemm (morphological analyzer for French ; F. Namer's website) ;

— part-of-speech taggers, with or without lemmatization : TreeTagger, Brill, *etc.* ;

— synonyms or paradigmatically related lexical units : WordNet (univ. Princeton or Java version at source.net/projects/jwordnet) ; the Roget's thesaurus, GREYC's dictionary of synonyms (Caen),*etc.* ;

— corpus-based paradigmatic relation acquisition, using non supervised machine learning techniques ; corpus-based semantic relation extraction using ILP, *etc.* ;

— complex term extraction : from French or English textual data, Acabit (B. Daille LINA Nantes), Ana (C. Enguehard LINA Nantes), Lexter (D. Bourigault ERSS Toulouse) and a more extended version Syntex

Don't forget that google is your friend.

7 What is expected at the end of the project

1. a search engine :
   — that is able to process a new QRY file that contains the complete set of queries (by the last practical session) ;
   — the search engine should be able to produce a file like the .REL one that contains 3 columns (query number, related document, similarity between both).
2. a report that summarize your work :
   — size and type of the vocabulary indexed ;
   — indexation method ;
   — similarity measure ;
   — tools used ;
   — *etc.*.