An Iterative Query Algorithm for Robust Systematic Review

Working Paper

Juste Raimbault

February 26th 2015

Abstract

Literature review is a crucial preliminary step for any scientific work and its quality and extent may have a dramatic impact on perspectives for research question and objectives. We propose an algorithm performing an automatized systematic review, to tackle in an original way this issue. Through iterative requests to a catalog and keyword extraction from the retrieven corpus, the final corpus ready for manual screening is built in a more robust way than with a single database request. We describe an implementation of the algorithm and show first results.

1 Introduction

Ignoring a significant contribution to a field or a particular theme when preparing a new related study is with recent technical means

2 Description of the Algorithm

Paradigm

Formalization Let $\mathcal{A}$ be an alphabet, $\mathcal{A}^*$ corresponding words and $\mathcal{T} = \cup_{n\in\mathbb{N}}\mathcal{A}^{*n}$ texts of finite length on it. A reference is for the algorithm a record with text fields representing title, abstract and keywords. Set of reference at iteration t will be denoted $R_t \in \mathcal{T}^3$

Convergence

3 Implementation

General Implementation Because of the heterogeneity of operations required by the algorithm (references organisation, catalog requests, text procesing), it was found a reasonable choice to implement it in Java. Source code and binaries are available on the Github repository of the project[1].

Catalog Requests Given a set of keywords, we need to extract a corpus of articles from a bibliographical database. It has to be done automatically and records must have abstract record populated as text mining is mostly done on it. Therefore, it was chosen to use Mendeley reference manager software [Mendeley, 2015] as it provide an open access to a flexible API that allows such catalog requests.

[1] at `https://github.com/JusteRaimbault/CityNetwork/tree/master/Models/Biblio/AlgoSR/AlgoSRJavaApp`

Natural Language Processing Keyword extraction is done through Natural Language Processing (NLP) techniques, following the workflow given in [Chavalarias and Cointet, 2013]. Although powerful and flexible libraries exist for current operations[2], the elaborated workflow of the paper would be painful to implement and is furthermore already made available by the authors on the dedicated website of the *CorText* project[3].

4 Results

References

[Chavalarias and Cointet, 2013] Chavalarias, D. and Cointet, J.-P. (2013). Phylomemetic patterns in science evolution—the rise and fall of scientific fields. *Plos One*, 8(2):e54847.

[Mendeley, 2015] Mendeley (2015). Mendeley reference manager. http://www.mendeley.com/.

[2] see e.g. Java library by The Stanford Natural Language Processing Group at `http://nlp.stanford.edu/software/corenlp.shtml`, or Python library NLTK at `http://www.nltk.org/`.

[3] `http://manager.cortext.net`