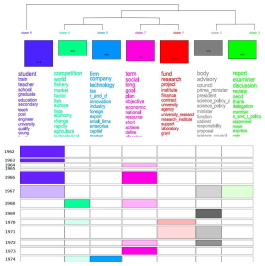
May 06, 2024

Pre-processing of a textual corpus for a lexicographic analysis with Iramuteq



This protocol is a draft, published without a DOI.

Wendeline Swart[1,2], Guillaume Cabanac[1,2,3], Cécile Crespy[4,2,3]

[1]Université Toulouse III – Paul Sabatier (IRIT); [2]Université de Toulouse; [3]Institut Universitaire de France; [4]Sciences Po Toulouse (LaSSP)

Wendeline Swart
Université Paul Sabatier (Toulouse III)

Protocol Citation: Wendeline Swart, Guillaume Cabanac, Cécile Crespy 2024. Pre-processing of a textual corpus for a lexicographic analysis with Iramuteq. **protocols.io** **https://protocols.io/view/pre-processing-of-a-textual-corpus-for-a-lexicogra-c4ybyxsn**

Protocol status: Working

Created: November 13, 2023

Last Modified: May 06, 2024

Protocol Integer ID: 90851

Keywords: lexicographic analysis, OECD reports

Abstract

This protocol presents a lexicographic analysis of a textual corpus composed of documents. In our case, we applied that method to a corpus of 41 **OECD** reports initially in PDF format. The metadata of each document (e.g., year, country) are considered as variables for the analysis. Among the lexicographic analysis software that are available, we picked Iramuteq, developed by LERASS, Université de Toulouse. The lexicographic analysis uncovers the topics of the corpus and their evolution. This protocol presents how to pre-process the corpus to avoid some pitfalls.

Image Attribution

Analysis of the corpus with Iramuteq.

Guidelines

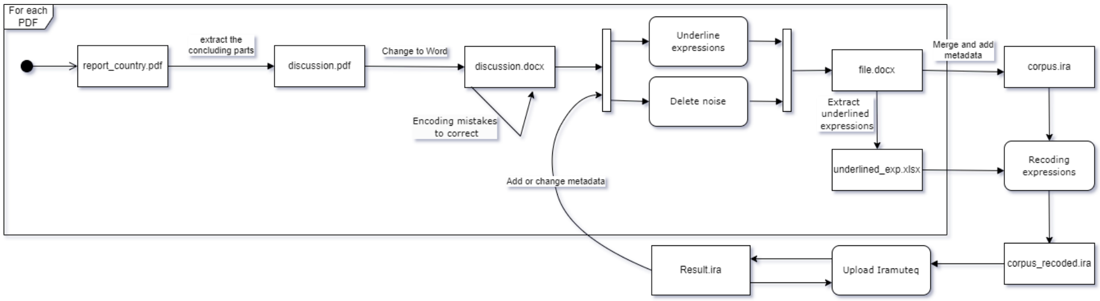
Warning: Iramuteq relies on a random variable when processing the text. For the same corpus the resulting analyses (e.g., classes) will not appear in the same order. This is because the program processes the text segments in a random order. The results produced by Iramuteq based on the same corpus are highly similar despite this random component.

Caveat: keep a version of the files produced at all steps to backtrack if necessary. This proved useful when re-coding the corpus.

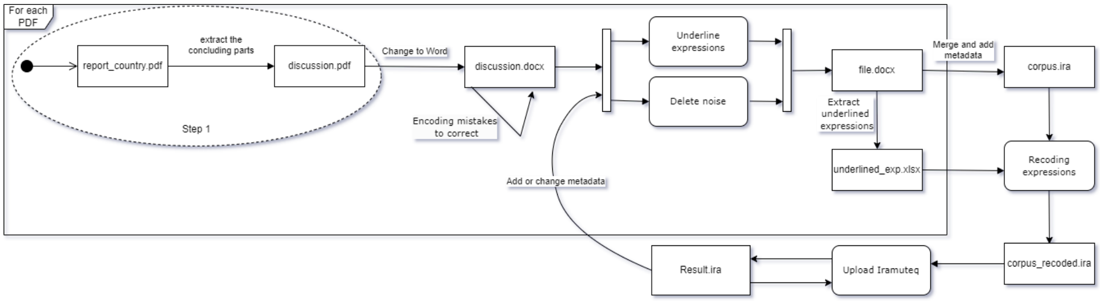
Materials

- Operating system: Windows Subsystem for Linux
- Software: Iramuteq (**http://www.iramuteq.org/**) version 3.
- Corpus: Scanned and OCRized science policy country reports published by the OECD between 1962 and 1996. The corpus was limited to the concluding parts of the 41 reports considered.

1 **Overall view of the protocol.** This protocol takes as input multiple documents in PDF and produces, as output, a single file containing the text to be analysed with the **Iramuteq** software.

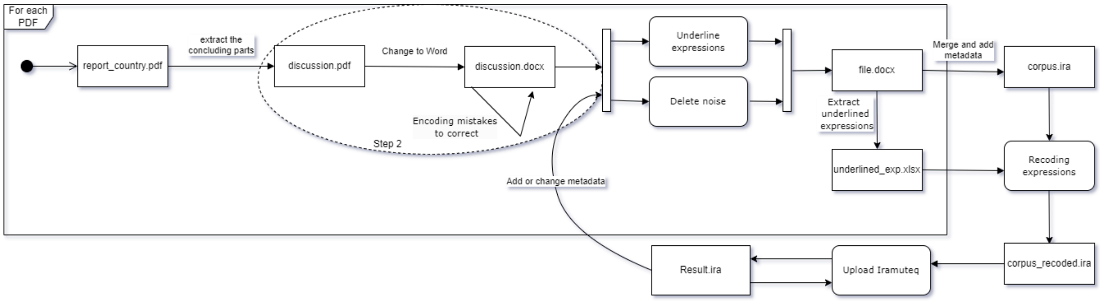


2 **Corpus delineation**. Open each PDF with Acrobat Reader Professional and extract the concluding parts (stored in discussion.pdf).



First step of the protocol.

3 **Reformatting the input files.**



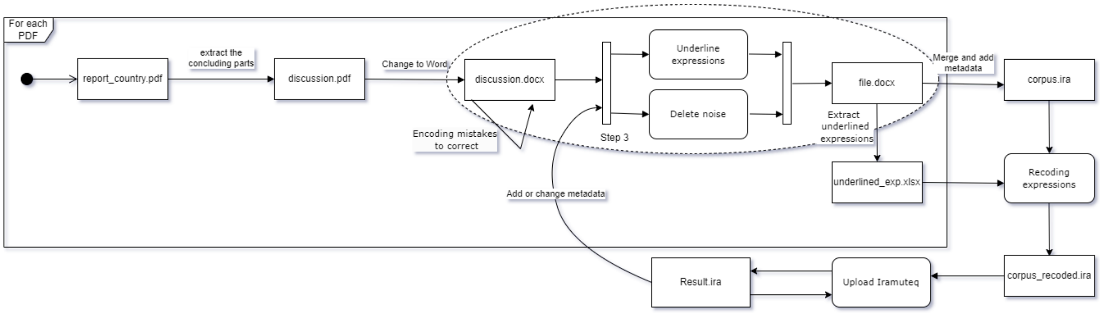
Second step of the protocol.

3.1 **Convert input files.** Open each PDF with Acrobat Reader Pro and convert into the Microsoft Word format (File, Save as… DOCX format).

3.2 **Correct OCR errors.** For each DOCX file, open it, spot the OCR errors (most of them are underlined in red) and correct them. Delete the passages that are not useful for a lexicometric analysis: headers, footers, and tables without relevant textual contents.

3.3 **Restore the paragraph structure.** Insert a newline between paragraphs to delineate sentences that must be considered together. Remove irrelevant newlines between sentences: some reflect page
breaks in the PDF.

4 **Manual pre-processing of the DOCX documents.**



Third step of the protocol.

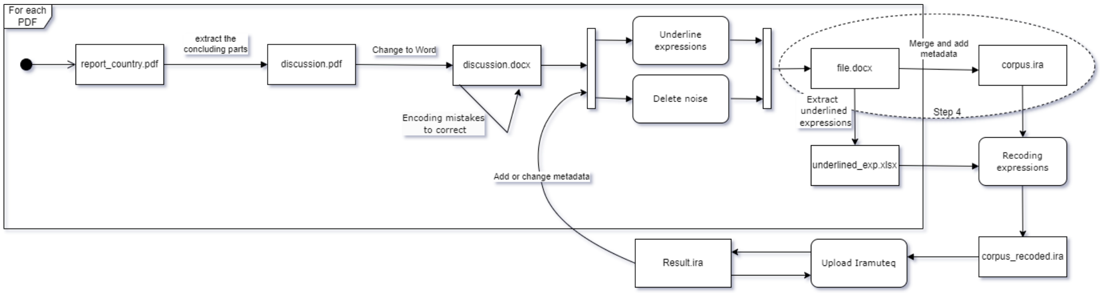
4.1 **Identify compound terms.** Compound terms are multi-word phrases that shouldn't be splitted during the analysis, such as "science policy." Underline all of them with the highlighting tool of Microsoft Word.

4.2 **Uniformise common terms.** identify the variations used for common terms in the corpus and recode variations to keep only one form. For instance, recode "Second World War", "2nd World War", "WW2" into "World_War_2" (note the underline character '_' instead of spaces). Another example: recode "European Union", "EU", "EEC", "Common Market" into "European_Union".

4.3 **Delete irrelevant terms.** Some terms convey no meaning for the lexicometric analysis. Remove them to prevent **Iramuteq** from inferring similarities between paragraphs only because these terms appear in several paragraphs. In the OECD reports, the name of the examiners are irrelevant (e.g., "Prof. X").

4.4 **Remove 'ego' in a document.** For a report on Country "X", remove all occurrences of "X" (i.e., ego) in the text. This prevents **Iramuteq** from making connections between paragraphs that do not use the same ideas but contain occurrences of the word "X" only.

5 **Automated merging** of the OCRised reports into a single **Iramuteq** corpus.



Fourth step of the protocol.

5.1 **Capitalise text.** Convert the text in uppercase to reduce homogeneity. You may use a text editor such as **Notepad++** or **BBEdit**.

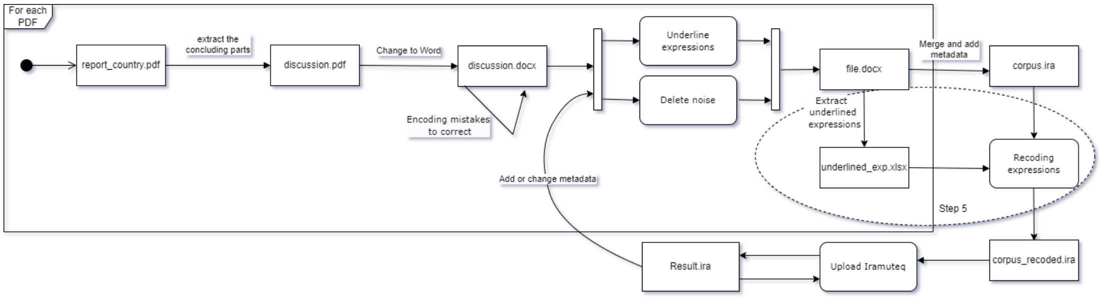
5.2 **Merge all documents.** Merge the DOCX files into a single DOCX, prefixing each text with the necessary **Iramuteq** metadata, inferred from the file names. The top line added reads as:

```
**** *variable1_value1 *variable2_value2 … *variableN_valueN
```

For instance, the 1966 report from France is prefixed by:

```
**** *COUNTRY_FRANCE *YEAR_1966
```

6 **Re-coding** of all variants of established phrases to prevent them from being split.



Fifth step of the protocol.

6.1 **Extract underlined expressions**. Run a Python script by parsing the HTML to extract all the underlined expressions. This script lists them in a single spreadsheet (e.g., an Excel file).

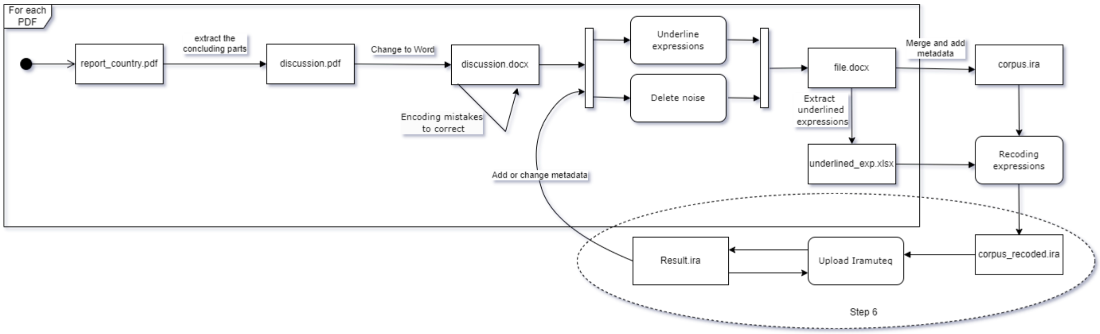
extract_underlined_words.py 2KB

6.2 **Delete duplicate expressions**. Delete duplicates that appear in the spreadsheet. Only one occurrence of each expression is needed for re-coding purposes.

6.3 **Sort the unique expressions** in alphabetical order. This allows the grouping of expressions referring
to a common topic (with a common start, e.g., EU and EUROPEAN_UNION).

6.4 **Standardise sentences**. The purpose of re-coding is to standardise similar expressions, such as "Science Policies" and its derivatives will become "SCIENCE_POLICY". To facilitate this step the column with the replacement value is automatically generated with spaces replaced by an underscore. The underscore character ('_') instructs **Iramuteq** to consider the phrase as a whole and prevent any splitting of the phrase into words. Consequently: change "SCIENCE POLICY" into "SCIENCE_POLICY."

7 **Produce the Iramuteq corpus.**



Sixth step of the protocol.

7.1 **Convert to TXT format**. Convert the Word document containing the entire corpus into TXT format. This is the format required by **Iramuteq**.

7.2 **Prioritise expressions to be re-coded**. This step replaces all expressions that have been re-coded manually in step 4.2. A Python script performs this substitution by replacing the longest expressions first. This avoids re-coding shorter expressions that would be included in larger ones, for example "COMMITTEE FOR SCIENCE POLICY" must be recoded as "CSTP" before "SCIENCE POLICY" is re-coded as "SCIENCE_POLICY" (otherwise, the first expression would become "COMMITTEE FOR CSTP", which is incorrect).

update_underlined_words.py 1KB

7.3 **Warning**. It is possible to repeat step 7.2 several times, when one realises that some extra expressions need to be recoded. In addition, one must pay attention to special characters in **Iramuteq**. For example, ampersands (&) are considered as spaces. For instance 'S&T' is indexed as the sequence 'S' followed by 'T', which is incorrect.

8 **Lexicometric analysis**. Once the re-coding is finished, open the TXT file with **Iramuteq**, selecting the language used in the corpus (English, for the OECD corpus). This protocol related to the *pre-processing* of a textual corpus ends here.