**List of files**

* Corpus : Corpus <- readtext("./AR.TXT/\*.pdf")
* Stop: stop words lists stop <- tibble(words = stopwords::data\_stopwords\_stopwordsiso$en)
* data.clean.no.punct: first cleaned file, no punctuation, no digits, double spaces removed. + changed the name of the pdf files for Annual Reports – year. 1 text per line
* data.clean.punct : same as the previous file but keeping the punctuation. 1 text per line
* tokenised.no.punct & tokenised.punct : data.clean.no.punct and data.clean.punct files are tokenised + removed from different pseudowords. (1 word per line)
* freq.million : tokenised.punct’s words are grouped by year of the document, counted and created a column with their frequency per million. Later the words are ungrouped.
* freq.million.and: from the freq.million file we have taken only the frequency per million of the word “and”.
* Freq.million.the: same as for the file freq.million.and but for the word “the”.
* tokenised.no.punct.nsw: to the files tokenised.no.punct and stop we have applied the anti\_join function of the package dplyr to get all the rows (where the words are contained) that do not have matching values. In other words, we have filtered the stop words from the tokenised.no.punct file.
* Later, to the tokenised.punct and tokenised.no.punct.nsw files we have applied a 4th cleaning layer.
* Dc.np.final : to the tokenised.no.punct.nsw file we have grouped the words by the year of the reports they belong to and do (summarise (text = paste(words, collapse= “”)). 1 text per line.
* Dc.p.final : to the tokenised.punct file we have grouped the words by the year of the reports they belong to and do (summarise (text = paste(words, collapse= “”)) 1 Text per line
* Wordnumperyear : to the tokenised.no.punct.nsw file we have grouped the words by annual report, counted their frequency and saved only the top 500 words.
* table.per.period : using the tokenised.no.punct.nsw file, we have divided the columns (years) in different periods and saved the words with a frequency of more than 10 times.
* Table.per.year: using the tokenised.no.punct.nsw file, we group its words by year, count them and save only those with a frequency superior to 15.
* Testtf: using the tokenised.no.punct.nsw file, we group its words by year, count them and save only those in the top 100 by frequency.

Data.clean.no.punct: 1 text per line

Tokenized.no.punct: 1 word per line

dc.np.final: 1 text per line

* pos.model: udpipe\_load\_model:
  + ftest:
  + POS.Rockefeller: we apply the POS analysis to the dc.p.final file, getting the POS of each word.