Package ‘rapidraker’

June 2, 2021

Type Package

Title Rapid Automatic Keyword Extraction (RAKE) Algorithm

Version 0.1.3

Description A 'Java' implementation of the RAKE algorithm ('Rose', S., 'Engel', D., 'Cramer', N. and 'Cowley', W. (2010) <doi:10.1002/9780470689646.ch1>), which can be used to extract keywords from documents without any training data.

URL https://crew102.github.io/slowraker/articles/rapidraker.html

BugReports https://github.com/crew102/rapidraker/issues

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 3.1)
Imports rJava, openNLPdata, slowraker, utils
Suggests knitr, rmarkdown, testthat

SystemRequirements Java (>= 8)

RoxygenNote 7.1.1

NeedsCompilation no

Author Christopher Baker [aut, cre]

Maintainer Christopher Baker <chriscrewbaker@gmail.com>

Repository CRAN

Date/Publication 2021-06-02 07:20:05 UTC

R topics documented:

   rapidrake ................................................................. 2

Index 4
Rapid RAKE

Description

A relatively fast version of the Rapid Automatic Keyword Extraction (RAKE) algorithm. See Automatic keyword extraction from individual documents for details on how RAKE works.

Usage

rapidrake(
  txt,
  stop_words = slowraker::smart_words,
  stop_pos = c("VB", "VBD", "VBG", "VBN", "VBP", "VBZ"),
  word_min_char = 3,
  stem = TRUE,
  phrase_delims = "[-,?\.:;"!/]
)

Arguments

txt A character vector, where each element of the vector contains the text for one document.

stop_words A vector of stop words which will be removed from your documents. The default value (smart_words) contains the 'SMART' stop words (equivalent to tm::stopwords('SMART')). Set stop_words = NULL if you don’t want to remove stop words.

stop_pos All words that have a part-of-speech (POS) that appears in stop_pos will be considered a stop word. stop_pos should be a vector of POS tags. All possible POS tags along with their definitions are in the pos_tags data frame (View(slowraker::pos_tags)). The default value is to remove all words that have a verb-based POS (i.e., stop_pos = c("VB", "VBD", "VBG", "VBN", "VBP", "VBZ"). Set stop_pos = NULL if you don’t want a word’s POS to matter during keyword extraction.

word_min_char The minimum number of characters that a word must have to remain in the corpus. Words with fewer than word_min_char characters will be removed before the RAKE algorithm is applied. Note that removing words based on word_min_char happens before stemming, so you should consider the full length of the word and not the length of its stem when choosing word_min_char.

stem Do you want to stem the words before running RAKE?

phrase_delims A regular expression containing the characters that will be used as phrase delimiters
Value

An object of class `rakelist`, which is just a list of data frames (one data frame for each element of `txt`). Each data frame will have the following columns:

- **keyword**  A keyword that was identified by RAKE.
- **freq**  The number of times the keyword appears in the document.
- **score**  The keyword’s score, as per the RAKE algorithm. Keywords with higher scores are considered to be higher quality than those with lower scores.
- **stem**  If you specified `stem = TRUE`, you will get the stemmed versions of the keywords in this column. When you choose stemming, the keyword’s score (`score`) will be based off its stem, but the reported number of times that the keyword appears (`freq`) will still be based off of the raw, unstemmed version of the keyword.

Examples

```r
## Not run:
rakelist <- rapidrake(txt = "some text that has great keywords")
slowraker::rbind_rakelist(rakelist)

## End(Not run)
```
Index

pos_tags, 2
rapidrake, 2