Package 'rtika'

January 21, 2018

Type Package

Title R Interface to 'Apache Tika'

Version 0.1.0

Author
Sasha Goodman <goodmansasha@gmail.com>[aut], The Apache Software Foundation [aut,cph]

Maintainer Sasha Goodman <goodmansasha@gmail.com>

Suggests sys, jsonlite, xml2

License Apache License 2.0 | file LICENSE

SystemRequirements Java (>=7) | openjdk-7-jre (via apt) | java-1.7.0-openjdk (via yum) | openjdk-8-jre (via apt) | java-1.8.0-openjdk (via yum)

Description Extract text and metadata from almost any file. Apache Tika parses over a thousand types, which is incredible but true. This R interface includes the Tika program. Read more at https://tika.apache.org/.

Depends R (>= 3.1.0)

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

URL http://github.com/predict-r/rtika

BugReports http://github.com/predict-r/rtika/issues

R topics documented:

tika *R Interface to 'Apache Tika'*

Description

Extract text and metadata from almost any file. Apache Tika parses over a thousand types, which is incredible but true. This R interface includes the Tika program. In addition to returning plain text, `rtika` can return partly structured text and metadata in `json`, `xml`, or `html`. For example, it will try to identify the `Content-Type` from pictures, videos, audio, code, and textual documents when `output="jsonRecursive"`, `output="xml"`, or `output="html"`. It automatically detects and parses several versions of Word, OpenOffice, rtf, iWorks, WordPerfect, pdf, epub, and more. It detects the character encodings of plain text files. It gets Exif from jpeg and tiff. It parses email mail boxes as well. See all the supported input formats here: https://tika.apache.org/1.17/formats.html.

Usage

```
tika(input_dir, output = c("text", "jsonRecursive", "xml", "html")[1],
  output_dir = "", n_chars = 1e+07, java = "java",
  jar = system.file("java", "tika-app-1.17.jar", package = "rtika"),
  threads = as.integer(1), args = character(), quiet = TRUE)
```

Arguments

`input_dir`	Directory where the files to be processed are. Each file in the directory will be read and analyzed but not changed.
`output`	Optional text format of the output. By default, `output = "text"`. That produces plain text without metadata. Use `output="jsonRecursive"` or `output="J"` to output metadata and content from the file and any embedded files, which can be parsed with the `jsonlite` package. Setting it to `output="xml"` or `output="x"` means the result of each file is XHTML, that can be parsed with other tools like the `XML` or `xml2` packages. The `output = "html"` or `output = "h"` is HTML, similar to XHTML.
`output_dir`	Optional directory path to save the result as files, as a side effect. Otherwise they are saved to a tmp directory R creates at startup and will be taken care of when R shuts down. Files are `.txt` by default, but can be `.json`, `.xml`, or `.html` depending on the `output` setting.
`n_chars`	Optional single integer specifying the maximum number of characters returned for each document, returned by the `readChar` function. The default is 1e+07. Higher numbers may be needed for exceptionally large files. There appears to be no advantage to lowering this, and no efficiency loss to raising it.
`java`	Optional alternative command to invoke Java. For example, it could be changed to the full path of a particular Java version. See the Configuration section below.
`jar`	Optional alternative path to the `tika-app-X.XX.jar`. Useful if the included version becomes out of date.
`threads`	Integer of the number of file consumer threads Tika uses. Defaults to 1.

args	Optional character vector of additional arguments for Tika, that are not yet implemented in this R interface, in the pattern of `c('-arg1','setting1','-arg2','setting2')`. Settable arguments include `-timeoutThresholdMillis` (Number of milliseconds allowed to a parse before the process is killed and restarted), `-maxRestarts` (Maximum number of times the watchdog process will restart the child process), `-includeFilePat` (Regular expression to determine which files to process, e.g. "(?i)\.pdf"), `-excludeFilePat`, and `-maxFileSizeBytes`. These are documented in the .jar –help command.
quiet	Logical if Tika command line messages and errors are to be supressed.

Value

A character vector, where each string corresponds to a file in the `input_dir`. The order is the same as that produced by `list.files(input_dir)`. If a file is not processed, the result will be NA. Also see the `output` options, above.

Background

Tika is a foundational library for several Apache projects, such as the Apache Solr search engine. This R interface produces a big payoff for R users. The most efficient way I've found to process tens of thousands of documents is Tika's 'batch' mode, which is used. There is more to do, given enough time and attention, because Apache Tika includes many other libraries and methods. The source is available at: https://tika.apache.org/.

Configuration

The first version of this package includes the `tika-app-X.XX.jar`. This jar works with Java 7. Tika in mid-2018 need Java 8. By default, this R package internally invokes Java by calling the `java` command from the command line. To change this, set the `java` attribute to call it another way (e.g. the full path to the location of a particular version of java).

Having the `sys` package is suggested but not required. `sys` speeds up calls to java and handles messaging much better. However, it requires the `libapparmor-dev` by default on unix variants, which must be installed before. Installing `sys` after `rtika` will work as well as installing it before.

Examples

```
# download file to some accessible directory
dir = file.path(getwd(),'tika-example');
dir.create(dir);
url = 'https://cran.r-project.org/doc/manuals/r-release/R-data.pdf'
download.file(url,file.path(dir,'R-data.pdf'))

#extract text
text = tika(dir)
cat(substr(text,1,2000))

#get metadata
if(requireNamespace('jsonlite')){
  json = tika(dir,'jsonRecursive')
```

```
  metadata = jsonlite::fromJSON(json[1])
  str(metadata) #data.frame of metadata

  metadata$'Content-Type' # [1] "application/pdf"
  metadata$producer # [1] "pdfTeX-1.40.18"
  metadata$'Creation-Date' # [1] "2017-11-30T13:39:02Z"
}
#don't forget to remove the downloaded test file
```

Index