# Package: tif (via r-universe)

July 12, 2024

**Type** Package

**Title** Text Interchange Format

**Version** 0.4

**Maintainer** Taylor B. Arnold <tarnold2@richmond.edu>

**Description** Provides validation functions for common interchange formats for representing text data in R. Includes formats for corpus objects, document term matrices, and tokens. Other annotations can be stored by overloading the tokens structure.

**Imports** Matrix

**License** GPL-2

**Encoding** UTF-8

**URL** https://docs.ropensci.org/tif, https://github.com/ropensci/tif

**BugReports** http://github.com/ropensci/tif/issues

**RoxygenNote** 7.2.1

**Suggests** testthat

**Repository** https://ropensci.r-universe.dev

**RemoteUrl** https://github.com/ropenscilabs/tif

**RemoteRef** master

**RemoteSha** 7a335fc5d620b41aa830abc7255f57674d8fa230

## Contents

tif-package . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
tif_as . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
tif_is_corpus_character . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
tif_is_corpus_df . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5
tif_is_dtm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
tif_is_tokens_df . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
tif_is_tokens_list . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

**Index** **9**

## tif-package *tif: Text Interchange Formats*

### Description

This package describes and validates formats for storing common object arising in text analysis as native R objects. Representations of a text corpus, document term matrix, and tokenized text are included. The corpus and tokens objects have multiple valid formats. Packages compliant with the tif proposal should accept all valid formats and should directly return, or provide conversion functions, for converting outputs into at least one of the formats (when applicable). The tokenized text format is extensible to include other annotations such as part of speech tags and named entities.

### Author(s)

**Maintainer**: Taylor Arnold <taylor.arnold@acm.org>

Authors:

- Ken Benoit <k.r.benoit@lse.ac.uk>
- Lincoln Mullen <lmullen@gmu.edu >
- Adam Obeng <contact@adamobeng.com>
- rOpenSci Text Workshop Participants (2017)

### See Also

Useful links:

- https://docs.ropensci.org/tif
- https://github.com/ropensci/tif
- Report bugs at http://github.com/ropensci/tif/issues

## tif_as *Coerce Between tif Object Specifications*

### Description

These functions convert between the various valid formats for corpus and tokens objects. By using these in other packages, maintainers need to only handle whichever specific format they would like to work with, but gain the freedom to output (or convert into) the one most suited to their package's paradigm.

### Usage

```
tif_as_corpus_character(corpus)

## Default S3 method:
tif_as_corpus_character(corpus)

## S3 method for class 'character'
tif_as_corpus_character(corpus)

## S3 method for class 'data.frame'
tif_as_corpus_character(corpus)

tif_as_corpus_df(corpus)

## Default S3 method:
tif_as_corpus_df(corpus)

## S3 method for class 'character'
tif_as_corpus_df(corpus)

## S3 method for class 'data.frame'
tif_as_corpus_df(corpus)

tif_as_tokens_df(tokens)

## Default S3 method:
tif_as_tokens_df(tokens)

## S3 method for class 'list'
tif_as_tokens_df(tokens)

## S3 method for class 'data.frame'
tif_as_tokens_df(tokens)

tif_as_tokens_list(tokens)

## Default S3 method:
tif_as_tokens_list(tokens)

## S3 method for class 'list'
tif_as_tokens_list(tokens)

## S3 method for class 'data.frame'
tif_as_tokens_list(tokens)
```

### Arguments

| | |
|---|---|
| `corpus` | valid tif corpus object to coerce |

| | |
|---|---|
| `tokens` | valid tif tokens object to coerce |

### Details

No explicit checking is done on the input; the output is guaranteed to be valid only if the input is a valid format. In fact, we make an effort to not modify an object that appears to be in the required format already due to R's copy on modify semantics.

### Examples

```
# coerce corpus object
corpus <- c("Aujourd'hui, maman est morte.",
            "It was a pleasure to burn.",
            "All this happened, more or less.")
names(corpus) <- c("Camus", "Bradbury", "Vonnegut")

new <- tif_as_corpus_df(corpus)
new
tif_as_corpus_character(new)

# coerce tokens object
tokens <- list(doc1 = c("aujourd'hui", "maman", "est", "morte"),
               doc2 = c("it", "was", "a", "pleasure", "to", "burn"),
               doc3 = c("all", "this", "happened", "more", "or", "less"))

new <- tif_as_tokens_df(tokens)
new
tif_as_tokens_list(new)
```

## tif_is_corpus_character

## *Validate Corpus Character Vector Object*

### Description

A valid character vector corpus object is an character vector with UTF-8 encoding. If it has names, this should be a unique character also in UTF-8 encoding. No other attributes should be present.

### Usage

```
tif_is_corpus_character(corpus, warn = FALSE)
```

### Arguments

| | |
|---|---|
| `corpus` | a corpus object to test for validity |
| `warn` | logical. Should the function produce a verbose warning for the condition for which the validation fails. Useful for testing. |

## Details

The tests are run sequentially and the function returns, with a warning if the warn flag is set, on the first test that fails. We use this implementation because some tests may fail entirely or be meaningless if the prior ones are note passed.

## Value

a logical vector of length one indicating whether the input is a valid corpus

## Examples

```
corpus <- c("Aujourd'hui, maman est morte.",
            "It was a pleasure to burn.",
            "All this happened, more or less.")

tif_is_corpus_character(corpus)

names(corpus) <- c("Camus", "Bradbury", "Vonnegut")
tif_is_corpus_character(corpus)
```

# tif_is_corpus_df *Validate Corpus Data Frame Object*

## Description

A valid data frame corpus object is an object that least two columns. One column must be called doc_id and be a character vector with UTF-8 encoding. Document ids must be unique. There must also be a column called text and must also be a character vector in UTF-8 encoding. Each individual document is represented by a single row in the data frame. Addition document-level metadata columns and corpus level attributes are allowed but not required.

## Usage

```
tif_is_corpus_df(corpus, warn = FALSE)
```

## Arguments

| | |
|---|---|
| corpus | a corpus object to test for validity |
| warn | logical. Should the function produce a verbose warning for the condition for which the validation fails. Useful for testing. |

## Details

The tests are run sequentially and the function returns, with a warning if the warn flag is set, on the first test that fails. We use this implementation because some tests may fail entirely or be meaningless if the prior ones are note passed. For example, if the corpus object does not have a variable named "text" it does not make sense to check whether this column is a character vector.

### Value

a logical vector of length one indicating whether the input is a valid corpus

### Examples

```
corpus <- data.frame(doc_id = c("doc1", "doc2", "doc3"),
                     text = c("Aujourd'hui, maman est morte.",
                      "It was a pleasure to burn.",
                      "All this happened, more or less."),
                     stringsAsFactors = FALSE)

tif_is_corpus_df(corpus)

corpus$author <- c("Camus", "Bradbury", "Vonnegut")
tif_is_corpus_df(corpus)
```

---

## tif_is_dtm *Validate Document Term Matrix Object*

---

### Description

A valid document term matrix is a sparse matrix with the row representing documents and columns representing terms. The row names is a character vector giving the document ids with no duplicated entries. The column names is a character vector giving the terms of the matrix with no duplicated entries. The spare matrix should inherit from the Matrix class dgCMatrix.

### Usage

```
tif_is_dtm(dtm, warn = FALSE)
```

### Arguments

`dtm` a document term matrix object to test the validity of

`warn` logical. Should the function produce a verbose warning for the condition for which the validation fails. Useful for testing.

### Details

The tests are run sequentially and the function returns, with a warning if the warn flag is set, on the first test that fails. We use this implementation because some tests may fail entirely or be meaningless if the prior ones are note passed. For example, if the dtm object is not a matrix it may not contain row or column names.

### Value

a logical vector of length one indicating whether the input is a valid document term matrix

## Examples

```
#' @importFrom Matrix Matrix
dtm <- Matrix::Matrix(0, ncol = 26, nrow = 5, sparse = TRUE)
colnames(dtm) <- LETTERS
rownames(dtm) <- sprintf("doc%d", 1:5)

tif_is_dtm(dtm)
```

---

# `tif_is_tokens_df` *Validate Tokens Data Frame Object*

---

## Description

A valid tokens data frame object is a data frame or an object that inherits a data frame. It has no row names and has at least two columns. It must a contain column called doc_id that is a character vector with UTF-8 encoding. Document ids must be unique. It must also contain a column called token that must also be a character vector in UTF-8 encoding. Each individual token is represented by a single row in the data frame. Addition token-level metadata columns are allowed but not required.

## Usage

```
tif_is_tokens_df(tokens, warn = FALSE)
```

## Arguments

| | |
|---|---|
| `tokens` | a tokens object to test for validity |
| `warn` | logical. Should the function produce a verbose warning for the condition for which the validation fails. Useful for testing. |

## Details

The tests are run sequentially and the function returns, with a warning if the warn flag is set, on the first test that fails. We use this implementation because some tests may fail entirely or be meaningless if the prior ones are note passed. For example, if the tokens object does not have a variable named "doc_id" it does not make sense to check whether this column is a character vector.

## Value

a logical vector of length one indicating whether the input is a valid tokens object

## Examples

```
tokens <- data.frame(doc_id = c("doc1", "doc1", "doc1", "doc1",
                                "doc2",  "doc2", "doc2", "doc2",
                                "doc2", "doc2", "doc3", "doc3",
                                "doc3", "doc3", "doc3", "doc3"),
                     token = c("aujourd'hui", "maman", "est",
                               "morte", "it", "was", "a", "pleasure",
                                    "to", "burn", "all", "this", "happened",
                                    "more", "or", "less"),
                          stringsAsFactors = FALSE)

tif_is_tokens_df(tokens)

tokens$pos <- "NOUN"
tokens$NER <- ""
tokens$sentiment <- runif(16L)
tif_is_tokens_df(tokens)
```

## tif_is_tokens_list *Validate Tokens List Object*

### Description

A valid corpus tokens object is (possibly named) list of character vectors. The character vectors, as well as names, should be in UTF-8 encoding. No other attributes should be present in either the list or any of its elements.

### Usage

```
tif_is_tokens_list(tokens, warn = FALSE)
```

### Arguments

| | |
|---|---|
| tokens | a tokens object to test for validity |
| warn | logical. Should the function produce a verbose warning for the condition for which the validation fails. Useful for testing. |

### Details

The tests are run sequentially and the function returns, with a warning if the warn flag is set, on the first test that fails. We use this implementation because some tests may fail entirely or be meaningless if the prior ones are note passed.

### Value

a logical vector of length one indicating whether the input is a valid tokens

### Examples

```
tokens <- list(doc1 = c("aujourd'hui", "maman", "est", "morte"),
               doc2 = c("it", "was", "a", "pleasure", "to", "burn"),
               doc3 = c("all", "this", "happened", "more", "or", "less"))
tif_is_tokens_list(tokens)

names(tokens) <- c("doc1", "doc2", "doc3")
tif_is_tokens_list(tokens)
```

# Index

tif (tif-package), 2
tif-package, 2
tif_as, 2
tif_as_corpus_character (tif_as), 2
tif_as_corpus_df (tif_as), 2
tif_as_tokens_df (tif_as), 2
tif_as_tokens_list (tif_as), 2
tif_is_corpus_character, 4
tif_is_corpus_df, 5
tif_is_dtm, 6
tif_is_tokens_df, 7
tif_is_tokens_list, 8