Package ‘tipitaka’

October 14, 2022

Type Package
Title Data and Tools for Analyzing the Pali Canon
Version 0.1.2
Description Provides access to the complete Pali Canon, or Tipitaka, the canonical scripture for Theravadin Buddhists worldwide. Based on the Chattha Sangayana Tipitaka version 4 (Vipassana Research Institute, 1990).
License CC0
Encoding UTF-8
LazyData true
LazyDataCompression xz
RoxygenNote 7.1.1
Depends R (>= 2.10)
Imports stringr, dplyr, magrittr, stringi, cpp11
LinkingTo cpp11
SystemRequirements C++14
NeedsCompilation yes
Author Dan Zigmond [aut, cre]
Maintainer Dan Zigmond <djz@shmonk.com>
Repository CRAN
Date/Publication 2021-03-30 11:10:02 UTC

R topics documented:

   abhidhamma_pitaka .................................................. 2
   pali_alphabet ....................................................... 3
   pali_eq ............................................................ 3
   pali_gt ............................................................. 4
   pali_lt ............................................................. 4
   pali_sort .......................................................... 5
abhidhamma_pitaka

pali_stop_words .................................................. 5
sati_sutta_long ................................................... 6
sati_sutta_raw ..................................................... 6
sutta_pitaka .......................................................... 7
tipitaka ............................................................... 8
tipitaka_long ......................................................... 9
tipitaka_names ...................................................... 9
tipitaka_raw .......................................................... 10
tipitaka_wide ......................................................... 10
vinaya_pitaka ........................................................ 11

Index 12

__________

abhidhamma_pitaka  All the books of the Abhidhamma Pitaka

Description

A subset of tipitaka_names consisting of only the books of the Abhidhamma Pitaka. These are easier to read if you call pali_string_fix() first.

Usage

abhidhamma_pitaka

Format

A tibble with the variables:

book  Abbreviated title
name  Full title

Examples

# Clean up the Unicode characters to make things more readable:
abhidhamma_pitaka$name <-
    stringi::stri_unescape_unicode(abhidhamma_pitaka$name)

# Count all the words in the Abhidhamma Pitaka:
sum(tipitaka_long[tipitaka_long$book %in% abhidhamma_pitaka$book, "n"])

pali_alphabet  

Pali alphabet in order

Description

Pali alphabet in order

Usage

pali_alphabet

Format

The Pali alphabet in traditional order.

Examples

# Returns TRUE because a comes before b in Pali:
match("a", pali_alphabet) < match("b", pali_alphabet)
# Returns FALSE because c comes before b in Pali
match("b", pali_alphabet) < match("c", pali_alphabet)

pali_eq  

Equal (==) comparison function for Pali words

Description

Note that all Pali string comparisons are case-insensitive.

Usage

pali_eq(word1, word2)

Arguments

word1  
A first Pali word as a string

word2  
A second Pali word as a string

Value

TRUE if word1 and word2 are the same
pali_gt  
*Greater-than (>)* comparison function for Pali words

**Description**

Note that all Pali string comparisons are case-insensitive. Also non-Pali characters are placed at the end of the alphabet and are considered equivalent to each other.

**Usage**

```pali_gt(word1, word2)```

**Arguments**

- `word1`: A first Pali word as a string
- `word2`: A second Pali word as a string

**Value**

TRUE if `word1` comes after `word2` alphabetically

---

pali_lt  
*Less-than (<)* comparison function for Pali words

**Description**

Note that all Pali string comparisons are case-insensitive. Also non-Pali characters are placed at the end of the alphabet and are considered equivalent to each other. This has been implemented in C++ for speed.

**Usage**

```pali_lt(word1, word2)```

**Arguments**

- `word1`: A first Pali word as a string
- `word2`: A second Pali word as a string

**Value**

TRUE if `word1` comes before `word2` alphabetically
pali_sort

Sorting function for vectors of Pali words.

Description

Note that all Pali string comparisons are case-insensitive. This algorithm is based on Quicksort, but creates lots of intermediate data structures instead of doing swaps in place. This has been implemented in C++ as the original R version was about 500x slower.

Usage

pali_sort(word_list)

Arguments

word_list

A vector of Pali words

Value

A new vector of Pali words in Pali alphabetical order

Examples

# Every unique word of the Mahāsatipatthāna Sutta in Pali alphabetical order:
word_list = pali_sort(sati_sutta_long$word)

# A sorted list of 100 random words from the Tipitaka:
library(dplyr)
word_list = sample(tipitaka_long$word, 100)

pali_stop_words

Tentative set of "stop words" for Pali

Description

A list of all declinables and particles from the PTS Pali-English Dictionary.

Usage

pali_stop_words

Format

An object of class tbl_df (inherits from tbl, data.frame) with 245 rows and 1 columns.
Source

https://dsalsrv04.uchicago.edu/dictionaries/pali/

Examples

# Find most common words in the Mahāsatipatthāna Sutta excluding stop words
library(dplyr)
sati_sutta_long %>%
  anti_join(pali_stop_words, by = "word") %>%
  arrange(desc(freq))

sati_sutta_long  Mahāsatipatthāna Sutta in "long" form

Description
The Mahāsatipatthāna Sutta or Discourse on the Establishing of Mindfulness in "long" form.

Usage

sati_sutta_long

Format
An object of class `data.frame` with 832 rows and 4 columns.

Source
Vipassana Research Institute, CST4, April 2020

sati_sutta_raw  Mahāsatipatthāna Sutta text in raw form

Description
The unprocessed text of the Mahāsatipatthāna Sutta

Usage

sati_sutta_raw

Format
A tibble with the variable:

 text  Complete text
sutta_pitaka

Source

Vipassana Research Institute, CST4, April 2020

sutta_pitaka  All the books of the Sutta Pitaka

Description

A subset of tipitaka_names consisting of only the books of the Sutta Pitaka. These are easier to read if you call stringi::stri_unescape_unicode first.

Usage

sutta_pitaka

Format

A tibble with the variables:

- **book**  Abbreviated title
- **name**  Full title

Examples

# Clean up the Unicode characters to make things more readable:
sutta_pitaka$name <-
  stringi::stri_unescape_unicode(sutta_pitaka$name)
# Count all the words in the Suttas:
sum(
  unique(
    tipitaka_long[tipitaka_long$book %in% sutta_pitaka$book, "total"]))

# Count another way:
sum(tipitaka_long[tipitaka_long$book %in% sutta_pitaka$book, "n"])

# Create a tibble of just the Suttas
sutta_wide <-
tipitaka_wide[row.names(tipitaka_wide) %in% sutta_pitaka$book,]
tipitaka

**Description**

The package tipitaka provides access to the complete Pali Canon, or Tipitaka, from R. The Tipitaka is the canonical scripture for Theravadin Buddhists worldwide. This version is largely taken from the Chāttha Sangāyana Tipitaka version 4.0 compiled by the Vispassana Research Institute, although edits have been made to conform to the numbering used by the Pali Text Society. This package provides both data and tools to facilitate the analysis of these ancient Pali texts.

**Data**

Several data sets are included:

- `tipitaka_raw`: the complete text of the Tipitaka
- `tipitaka_long`: the complete Tipitaka in "long" form
- `tipitaka_wide`: the complete Tipitaka in "wide" form
- `tipitaka_names`: the names of each book of the Tipitaka
- `sutta_pitaka`: the names of each volume of the Sutta Pitaka
- `vinaya_pitaka`: the names of each volume of the Vinaya Pitaka
- `abhidhamma_pitaka`: the names of each volume of the Abhidhamma Pitak
- `sati_sutta_raw`: the Mahāsatipatthāna Sutta text
- `sati_sutta_long`: the Mahāsatipatthāna Sutta in "long" form
- `pali_alphabet`: the complete pali alphabet in traditional order
- `pali_stop_words`: a set of "stop words" for Pali

**Tools**

A few useful functions are provided for working with Pali text:

- `pali_lt`: less-than function for Pali strings
- `pali_gt`: greater-than function for Pali strings
- `pali_eq`: equals function for Pali strings
- `pali_sort`: sorting function for vectors of pali strings
tipitaka_long

Tipitaka in "long" form

Description
Every word of every volume of the Tipitaka, with one word per volume per line.

Usage
tipitaka_long

Format
A tibble with the variables:

- **word**: Pali word
- **n**: Number of time this word appears in this book
- **total**: Total number of words in this book
- **freq**: Frequency with which this word appears in this book
- **book**: Abbreviated book name

Source
Vipassana Research Institute, CST4, April 2020

tipitaka_names

Names of each book of the Tipitaka, both abbreviated and in full. These are easier to read if you call `pali_string_fix()` first.

Description
Names of each book of the Tipitaka, both abbreviated and in full. These are easier to read if you call `pali_string_fix()` first.

Usage
tipitaka_names

Format
A tibble with the variables:

- **book**: Abbreviated title
- **name**: Full title
Examples

```r
# Clean up the Unicode characters to make things more readable:
tipitaka_names$name <-
  stringi::stri_unescape_unicode(tipitaka_names$name)
```

<table>
<thead>
<tr>
<th>tipitaka_raw</th>
<th>Tipitaka text in raw form</th>
</tr>
</thead>
</table>

Description

The unprocessed text of the Tipitaka, with one row per volume.

Usage

tipitaka_raw

Format

A tibble with the variables:

- **text**: Text of each Tipitaka volume
- **book**: Abbreviated book name of each volume

Source

Vipassana Research Institute, CST4, April 2020

<table>
<thead>
<tr>
<th>tipitaka_wide</th>
<th>Tipitaka in &quot;wide&quot; form</th>
</tr>
</thead>
</table>

Description

Every word of every volume of the Tipitaka, with one word per column and one book per line. Each cell is the frequency at which that word appears in that book.

Usage

tipitaka_wide

Format

An object of class `data.frame` with 46 rows and 141360 columns.

Source

Vipassana Research Institute, CST4, April 2020
Description

A subset of tipitaka_names consisting of only the books of the Vinaya Pitaka. These are easier to read if you call stringi::stri_unescape_unicode first.

Usage

vinaya_pitaka

Format

A tibble with the variables:

- book  Abbreviated title
- name  Full title

Examples

# Clean up the Unicode characters to make things more readable:
vinaya_pitaka$name <- stringi::stri_unescape_unicode(vinaya_pitaka$name)

# Count all the words in the Vinaya Pitaka:
sum(tipitaka_long[tipitaka_long$book %in% vinaya_pitaka$book, "n"])

---

vinaya_pitaka  All the books of the Vinaya Pitaka
Index

* datasets
  abhidhamma_pitaka, 2
  pali_alphabet, 3
  pali_stop_words, 5
  sati_sutta_long, 6
  sati_sutta_raw, 6
  sutta_pitaka, 7
  tipitaka_long, 9
  tipitaka_names, 9
  tipitaka_raw, 10
  tipitaka_wide, 10
  vinaya_pitaka, 11

abhidhamma_pitaka, 2
pali_alphabet, 3
pali_eq, 3
pali_gt, 4
pali_lt, 4
pali_sort, 5
pali_stop_words, 5
sati_sutta_long, 6
sati_sutta_raw, 6
sutta_pitaka, 7

tipitaka, 8
tipitaka_long, 9
tipitaka_names, 9
tipitaka_raw, 10
tipitaka_wide, 10

vinaya_pitaka, 11