

# Package ‘cld2’

December 15, 2020

**Type** Package

**Title** Google's Compact Language Detector 2

**Version** 1.2.1

**Description** Bindings to Google's C++ library Compact Language Detector 2 (see <https://github.com/cld2owners/cld2#readme> for more information). Probabilistically detects over 80 languages in plain text or HTML. For mixed-language input it returns the top three detected languages and their approximate proportion of the total classified text bytes (e.g. 80% English and 20% French out of 1000 bytes). There is also a 'cld3' package on CRAN which uses a neural network model instead.

**License** Apache License 2.0

**Encoding** UTF-8

**LazyData** true

**URL** <https://docs.ropensci.org/cld2/> (docs)  
<https://github.com/ropensci/cld2> (devel)  
<https://github.com/cld2owners/cld2> (upstream)

**Imports** Rcpp

**LinkingTo** Rcpp

**RoxygenNote** 6.0.1

**Suggests** testthat, readtext, cld3

**NeedsCompilation** yes

**Author** Jeroen Ooms [aut, cre] (<https://orcid.org/0000-0002-4035-0289>),  
Dirk Sijtsma [cph] (Author of CLD2 C++ library)

**Maintainer** Jeroen Ooms <[jeroen@berkeley.edu](mailto:jeroen@berkeley.edu)>

**Repository** CRAN

**Date/Publication** 2020-12-15 22:50:09 UTC

## R topics documented:

cld2	2
Index	3

**Description**

The function `detect_language()` is vectorised and guesses the the language of each string in text or returns NA if the language could not reliably be determined. The function `detect_language_multi()` is not vectorised and analyses the entire character vector as a whole. The output includes the top 3 detected languages including the relative proportion and the total number of text bytes that was reliably classified.

**Usage**

```
detect_language(text, plain_text = TRUE, lang_code = TRUE)
```

```
detect_language_mixed(text, plain_text = TRUE)
```

**Arguments**

<code>text</code>	a string with text to classify or a connection to read from
<code>plain_text</code>	if FALSE then code skips HTML tags and expands HTML entities
<code>lang_code</code>	return a language code instead of name

**Examples**

```
# Vectorized function
text <- c("To be or not to be?", "Ce n'est pas grave.", "Nou breekt mijn klomp!")
detect_language(text)

## Not run:
# Read HTML from connection
detect_language(url('http://www.un.org/ar/universal-declaration-human-rights/'), plain_text = FALSE)

# More detailed classification output
detect_language_mixed(
  url('http://www.un.org/fr/universal-declaration-human-rights/'), plain_text = FALSE)

detect_language_mixed(
  url('http://www.un.org/zh/universal-declaration-human-rights/'), plain_text = FALSE)

## End(Not run)
```

# Index

`cld2`, [2](#)

`detect_language(cld2)`, [2](#)

`detect_language()`, [2](#)

`detect_language_mixed(cld2)`, [2](#)

`detect_language_multi(cld2)`, [2](#)

`detect_language_multi()`, [2](#)