Package ‘cidian’

September 28, 2026

Title Read and Parse Chinese Input-Method Dictionaries

Version 0.1.0

Description Read Chinese input-method dictionary files into a common 'R' data model. The 'Rust' backend supports Sogou, QQ Pinyin, and Baidu dictionary formats and preserves source metadata, code components, and weights. This is useful for building a custom Chinese word segmentation dictionary.

License MIT + file LICENSE

URL https://github.com/Yousa-Mirage/r-cidian

BugReports https://github.com/Yousa-Mirage/r-cidian/issues

Encoding UTF-8

Config/rextendr/version 0.5.0

SystemRequirements Cargo (Rust's package manager), rustc >= 1.85.0, xz

Depends R (>= 4.2)

Imports cli, rlang

Config/roxygen2/version 8.1.0

Suggests rmarkdown, spelling, testthat (>= 3.0.0)

Config/testthat/edition 3

Config/testthat/parallel true

Language en-US

NeedsCompilation yes

Author Hao Cheng [aut, cre, cph]

Maintainer Hao Cheng <Yousa-Mirage@foxmail.com>

Repository CRAN

Date/Publication 2026-09-28 13:40:09 UTC

Contents

`as.data.frame.cidian_dictionary`
Convert a dictionary to a data frame

Description

Convert a dictionary to a data frame

Usage

```
## S3 method for class 'cidian_dictionary'
as.data.frame(x, ...)
```

Arguments

`x`	A `cidian_dictionary` object.
`...`	Must be empty.

Value

A base `data.frame` with `word`, `code`, and `weight` columns. `code` is a list-column of character vectors and `weight` is numeric.

Examples

```
x <- read_cidian(system.file("extdata", "computer.qcel", package = "cidian"))
head(as.data.frame(x))
```

cidian_entries *Extract dictionary entries*

Description

Extract dictionary entries

Usage

```
cidian_entries(x)
```

Arguments

x	A cidian_dictionary object.

Value

A base data.frame with word, code, and weight columns. code is a list-column of character vectors and weight is numeric.

Examples

```
x <- read_cidian(system.file("extdata", "computer.qcel", package = "cidian"))
head(cidian_entries(x))
```

cidian_formats *List supported dictionary formats*

Description

List supported dictionary formats

Usage

```
cidian_formats()
```

Value

A character vector of format identifiers accepted by read_cidian().

Examples

```
cidian_formats()
```

cidian_metadata *Extract dictionary metadata*

Description

Extract dictionary metadata

Usage

```
cidian_metadata(x)
```

Arguments

x	A cidian_dictionary object.

Value

A list with name, category, description, and extra fields.

Examples

```
x <- read_cidian(system.file("extdata", "computer.qcel", package = "cidian"))
cidian_metadata(x)
```

print.cidian_dictionary *Print a dictionary summary*

Description

Print a dictionary summary

Usage

```
## S3 method for class 'cidian_dictionary'
print(x, ...)
```

Arguments

x	A cidian_dictionary object.
...	Must be empty.

Value

Invisibly returns a summary.cidian_dictionary object summarizing x.

Examples

```
x <- read_cidian(system.file("extdata", "computer.qcel", package = "cidian"))
print(x)
```

`read_cidian` *Read a Chinese input-method dictionary*

Description

`read_cidian()` reads a supported binary dictionary and returns a common `cidian_dictionary` object. The parser preserves source order and does not normalize, sort, or deduplicate entries.

Usage

```
read_cidian(path, ..., format = c("scel", "qcel", "qpyd", "bdict", "bcd"))
```

Arguments

`path`	A path to a dictionary file.
`...`	Must be empty.
`format`	A format name, with or without a leading dot. Supported values are `"scel"`, `"qcel"`, `"qpyd"`, `"bdict"`, and `"bcd"`.

Details

The format is inferred from the file extension when `format` is omitted or `NULL`. Supply `format` when the file has no extension or its extension is wrong.

Value

A `cidian_dictionary` object containing `metadata`, `entries`, and `format` fields. The `entries$code` column is a list-column of character vectors, and `entries$weight` is a numeric vector.

Examples

```
x <- read_cidian(system.file("extdata", "computer.qcel", package = "cidian"))
x
```

summary.cidian_dictionary *Summarize a dictionary*

Description

Summarize a dictionary

Usage

```
## S3 method for class 'cidian_dictionary'
summary(object, ...)
```

Arguments

object	A cidian_dictionary object.
...	Must be empty.

Value

An object of class summary.cidian_dictionary containing the format, metadata, entry count, and number of weighted entries.

Examples

```
x <- read_cidian(system.file("extdata", "computer.qcel", package = "cidian"))
summary(x)
```

write_words *Write dictionary entries to a text file*

Description

write_words() writes the word of every entry in x to path, one word per line, preserving the source order.

Usage

```
write_words(x, path, ..., overwrite = NULL)
```

Arguments

x	A cidian_dictionary object.
path	A path to the output file.
...	Must be empty.
overwrite	Whether to replace path when it already exists. NULL (the default) asks interactively; TRUE overwrites without asking; FALSE errors.

Details

When `path` already exists, the function asks for confirmation unless `overwrite` is given explicitly.

Value

The `path`, invisibly.

Examples

```
x <- read_cidian(system.file("extdata", "computer.qcel", package = "cidian"))
write_words(x, tempfile(fileext = ".txt"))
```

Index