

# Package ‘lang’

October 30, 2025

**Title** Translates R Help Documentation using Large Language Models

**Version** 0.1.0

**Description** Translates R help documentation on the fly by using a Large Language model of your choice. If you are using 'RStudio' or 'Positron' the translated help will appear in the help pane.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Imports** callr, cli, fs, glue, mall, rlang, rstudioapi, tools, withr

**Suggests** testthat (>= 3.0.0), ellmer

**Config/testthat/edition** 3

**URL** <https://mlverse.github.io/lang/>

**NeedsCompilation** no

**Author** Edgar Ruiz [aut, cre]

**Maintainer** Edgar Ruiz <edgar@posit.co>

**Repository** CRAN

**Date/Publication** 2025-10-30 20:10:02 UTC

## Contents

help . . . . .	2
lang_help . . . . .	2
lang_use . . . . .	3
<b>Index</b>	<b>5</b>

---

help

*Drop-in replacements for help and ? functions*


---

### Description

The `?` and `help` functions are replacements for functions of the same name in the `utils` package. If the `LANG` environment variable is not set to English, it will activate the translation to whatever language `LANG` is set to.

### Usage

```
# help(topic, package = NULL, ...)

# ?e2
# e1?e2
```

### Arguments

<code>topic</code>	A name or character string specifying the help topic.
<code>package</code>	A name or character string specifying the package in which to search for the help topic. If <code>NULL</code> , search all packages.
<code>...</code>	Additional arguments to pass to <code>utils::help()</code> .
<code>e1</code>	First argument to pass along to <code>utils::?</code> .
<code>e2</code>	Second argument to pass along to <code>utils::?</code> .

---

lang\_help

*Translates help documentation to another language*


---

### Description

Translates a given topic into a target language. It uses the `lang` argument to determine which language to translate to. If not passed, this function will look for a target language in the `LANG` and `LANGUAGE` environment variables, or if something has been passed to the `.lang` argument in `lang_use()`, to determine the target language. If the target language is English, no translation will be processed, so the help returned will be the original package's documentation.

### Usage

```
lang_help(topic, package = NULL, lang = NULL, type = getOption("help_type"))
```

**Arguments**

topic	A character vector of the topic to search for.
package	The R package to look for the topic, if not provided the function will attempt to find the topic based on the loaded packages.
lang	A character vector language to translate the topic to
type	Produce "html" or "text" output for the help. It defaults to <code>getOption("help_type")</code>

**Value**

Original or translated version of the help documentation in the output type specified

**Examples**

```
library(lang)

lang_use("ollama", "llama3.2", seed = 100)

lang_help("lang_help", lang = "spanish", type = "text")
```

---

lang_use	<i>Specifies the LLM provider and model to use during the R session</i>
----------	---

---

**Description**

Allows us to specify the back-end provider, model to use during the current R session. The target language is not processed by the function, as in converting "english" to "en" for example. The value is passed directly to the LLM, and it lets the LLM interpret the target language.

**Usage**

```
lang_use(
  backend = NULL,
  model = NULL,
  .cache = NULL,
  .lang = NULL,
  .silent = FALSE,
  ...
)
```

**Arguments**

backend	"ollama" or an <code>ellmer Chat</code> object. If using "ollama", <code>mall</code> will use is out-of-the-box integration with that back-end. Defaults to "ollama".
model	The name of model supported by the back-end provider

<code>.cache</code>	The path to save model results, so they can be re-used if the same operation is ran again. To turn off, set this argument to an empty character: <code>""</code> . It defaults to a temp folder. If this argument is left NULL when calling this function, no changes to the path will be made.
<code>.lang</code>	Target language to translate to. This will override values found in the LANG and LANGUAGE environment variables.
<code>.silent</code>	Boolean flag that controls if there is or not output to the console. Defaults to FALSE.
<code>...</code>	Additional arguments that this function will pass down to the integrating function. In the case of Ollama, it will pass those arguments to <code>ollamar::chat()</code> .

### Value

Console output of the current LLM setup to be used during the R session.

### Examples

```
library(lang)

# Using an `ellmer` chat object
lang_use(ellmer::chat_openai(model = "gpt-4o"))

# Using Ollama directly
lang_use("ollama", "llama3.2", seed = 100)

# Turn off cache by setting `.cache` to ""
lang_use("ollama", "llama3.2", seed = 100, .cache = "")

# Use `.lang` to set the target language to translate to,
# it will be set for the current R session
lang_use("ollama", "llama3.2", .lang = "spanish")

# Use `.silent` to avoid console output
lang_use("ollama", "llama3.2", .lang = "spanish", .silent = TRUE)

# To see current settings, simply call the function
lang_use()
```

# Index

? (help), [2](#)

help, [2](#)

lang\_help, [2](#)

lang\_use, [3](#)

shim\_lang\_help (help), [2](#)

shim\_lang\_question (help), [2](#)

utils::help(), [2](#)