Package: codestral (via r-universe)

October 13, 2026

Title Chat and FIM with 'Codestral'

Version 0.0.2.9001

Date 2026-01-13

Description Create an addin in 'Rstudio' to do fill-in-the-middle (FIM) and chat with latest Mistral AI models for coding, 'Codestral' and 'Codestral Mamba'. For more details about 'Mistral AI API': <https://docs.mistral.ai/getting-started/quickstart/> and <https://docs.mistral.ai/api/>. For more details about 'Codestral' model: <https://mistral.ai/news/codestral>; about 'Codestral Mamba': <https://mistral.ai/news/codestral-mamba>.

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports dplyr, httr, jsonlite, magrittr, rstudioapi, stringr, utils

Depends R (>= 4.1)

LazyData true

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

URL https://urbs-dev.github.io/codestral/

BugReports https://github.com/urbs-dev/codestral/issues

VignetteBuilder knitr

License MIT + file LICENSE

Config/pak/sysreqs libicu-dev libssl-dev

Repository https://urbs-dev.r-universe.dev

Date/Publication 2026-10-13 14:19:32 UTC

RemoteUrl https://github.com/urbs-dev/codestral

RemoteRef HEAD

RemoteSha fdf3d05e2475b93d0f0e869868db445484127d74

Contents

allow_detect_package . . . 2
codestral . . . 3
codestral_init . . . 4
compile_dialog . . . 5
complete_current_script . . . 6
debug_mode . . . 6
detect_package . . . 7
ENDPOINTS . . . 7
include_file . . . 7
include_package_files . . . 8
insert_addin . . . 9
inventory_Rfiles . . . 9
run_prompt . . . 10

Index **11**

`allow_detect_package` *Allow codestral to detect package environment*

Description

Allow codestral to detect package environment

Usage

```
allow_detect_package(x = TRUE)
```

Arguments

`x`	A boolean indicating that the user allows the detection of a package environment. Defaulted to `TRUE`.

Details

If set to `TRUE`, when codestral is used in a folder with a `DESCRIPTION` file, all R files in the current folder and its subfolders are included in the prompt.

Value

`0` invisible.

codestral *Fill in the middle with Codestral*

Description

This function completes a given prompt using the Codestral API. It supports different models for fill-in-the-middle, chat with Codestral, and chat with Codestral Mamba. The function relies on environment variables for some parameters.

Usage

```
codestral(
  prompt,
  suffix = "",
  path = NULL,
  mistral_apikey = Sys.getenv(x = "R_MISTRAL_APIKEY"),
  codestral_apikey = Sys.getenv(x = "R_CODESTRAL_APIKEY"),
  fim_model = Sys.getenv(x = "R_CODESTRAL_FIM_MODEL"),
  chat_model = Sys.getenv(x = "R_CODESTRAL_CHAT_MODEL"),
  mamba_model = Sys.getenv(x = "R_MAMBA_CHAT_MODEL"),
  temperature = as.integer(Sys.getenv(x = "R_CODESTRAL_TEMPERATURE")),
  max_tokens_FIM = Sys.getenv(x = "R_CODESTRAL_MAX_TOKENS_FIM"),
  max_tokens_chat = Sys.getenv(x = "R_CODESTRAL_MAX_TOKENS_CHAT"),
  role_content = Sys.getenv(x = "R_CODESTRAL_ROLE_CONTENT")
)
```

Arguments

prompt	The prompt to complete.
suffix	The suffix to use. Defaults to an empty string.
path	The path to the current file. Defaults to NULL.
mistral_apikey, codestral_apikey	The API keys to use for accessing Codestral Mamba and Codestral. Default to the value of the R_MISTRAL_APIKEY, R_CODESTRAL_APIKEY environment variable. Note that the name of the variable mistra_apikey is purposely not mentionning Codestral Mamba because this key can be use for other Mistral AI models (except Codestral).
fim_model	The model to use for fill-in-the-middle. Defaults to the value of the R_CODESTRAL_FIM_MODEL environment variable.
chat_model	The model to use for chat with Codestral. Defaults to the value of the R_CODESTRAL_CHAT_MODEL environment variable.
mamba_model	The model to use for chat with Codestral Mamba. Defaults to the value of the R_MAMBA_CHAT_MODEL environment variable.
temperature	The temperature to use. Defaults to the value of the R_CODESTRAL_TEMPERATURE environment variable.

max_tokens_FIM, max_tokens_chat
: Integers giving the maximum number of tokens to generate for FIM and chat. Defaults to the value of the R_CODESTRAL_MAX_TOKENS_FIM, R_CODESTRAL_MAX_TOKENS_CHAT environment variables.

role_content
: The role content to use. Defaults to the value of the R_CODESTRAL_ROLE_CONTENT environment variable.

Value

A character string containing the completed text.

---

codestral_init *Initialize codestral*

Description

Create environment variables for operationg FIM and chat.

Usage

```
codestral_init(
  mistral_apikey = Sys.getenv(x = "R_MISTRAL_APIKEY"),
  codestral_apikey = Sys.getenv(x = "R_CODESTRAL_APIKEY"),
  fim_model = "codestral-latest",
  chat_model = "codestral-latest",
  mamba_model = "open-codestral-mamba",
  temperature = 0,
  max_tokens_FIM = 100,
  max_tokens_chat = "",
  role_content = NULL
)
```

Arguments

mistral_apikey, codestral_apikey
: The API keys to use for accessing Codestral Mamba and Codestral. Default to the value of the R_MISTRAL_APIKEY, R_CODESTRAL_APIKEY environment variable. Note that the name of the variable mistra_apikey is purposely not mentionning Codestral Mamba because this key can be use for other Mistral AI models (except Codestral).

fim_model
: A string giving the model to use for FIM.

chat_model
: A string giving the model to use for Codestral chat.

mamba_model
: A string giving the model to use for Codestral Mamba chat.

temperature
: An integer giving the temperature to use.

max_tokens_FIM, max_tokens_chat
: Integers giving the maximum number of tokens to generate for each of these operations.

`role_content` A role to assign to the system Default is "You write programs in R language only. You adopt a proper coding approach by strictly naming all the functions' parameters when calling any function with named parameters even when calling nested functions, by being straighforward in your answers."

Details

The most important paremeters here are the `..._apikey` parameters without which the Mistral AI API can not be used.

To start with, beginners may keep default values for other parameters. It seems sound to use the latest models of each type. However with time, the user may be willing to customize `temperature`, `max_tokens_FIM`, `max_tokens_chat` and `role_content` for his/her own needs.

This function creates the following environment variables using `Sys.setenv()`:

- R_MISTRAL_APIKEY
- R_CODESTRAL_APIKEY
- R_CODESTRAL_FIM_MODEL
- R_CODESTRAL_CHAT_MODEL
- R_CODESTRAL_MAMBA_MODEL
- R_CODESTRAL_TEMPERATURE
- R_CODESTRAL_MAX_TOKENS_FIM
- R_CODESTRAL_MAX_TOKENS_CHAT
- R_CODESTRAL_DEBUG
- R_CODESTRAL_DETECT_PACKAGE
- R_CODESTRAL_ROLE_CONTENT

Value

Invisible `0`.

`compile_dialog` *Analyses a prompt to re-buid the dialog*

Description

Analyses a prompt to re-buid the dialog

Usage

```
compile_dialog(prompt)
```

Arguments

`prompt` The prompt to analyse. A vector of strings.

Value

A list with the chatter (Codestral or Codestral Mamba) and the dialog in a data.frame whith columns `role` and `content`.

---

complete_current_script *Fill in the middle or complete*

Description

This function splits the current script into two parts: the part before the cursor and the part after the cursor.

Usage

```
complete_current_script()
```

Value

A list with elements `ans` (character), `doc_id` (document ID) and `location` (document_position).

---

debug_mode *Set debug mode*

Description

Set debug mode

Usage

```
debug_mode(debug = TRUE)
```

Arguments

debug	A logical value. If `TRUE`, the debug mode is activated.

Details

When the debug mode is activated, the function `codestral()` will print the request body and the response body.

Value

Invisible `0`.

detect_package *Detect whether the working directory is that of a package*

Description

Detect whether the working directory is that of a package

Usage

```
detect_package()
```

Value

TRUE if the working directory is that of a package, FALSE otherwise

ENDPOINTS *Endpoints for the Codestral API.*

Description

Endpoints for the Codestral API.

Usage

```
ENDPOINTS
```

Format

A named list with elements chat and completion.

include_file *Read and include files in a prompt*

Description

Read and include files in a prompt

Usage

```
include_file(prompt, anyFile)
```

Arguments

prompt	A vector of strings.
anyFile	A boolean of the same length of prompt indicating that an instruction "ff:" has been detected in a string.

Details

If anyFile[i] is TRUE then the sequence of characters following the instruction "ff:" in prompt[i] is read until the next space or the end of the string. This extracted string is assumed to be a file name. This file is looked for in the current working directory or any of its sub-directories. Once detected, the file is read with readLines() and this content is inserted in prompt between prompt[i-1] and prompt[i+1]. Note that prompt[i] is therefore deleted.

The result is returned.

Value

A vector of strings containing prompt augmented by the files refered to in the original prompt.

---

include_package_files *Include package files in the prompt*

Description

This function includes R files from a package in the prompt when a package is detected. It handles the inclusion of file contents and path-specific exclusions.

Usage

```
include_package_files(prompt, path)
```

Arguments

prompt	The current prompt to which package files should be added.
path	Current file path to exclude from the included files.

Value

The modified prompt with package file contents included.

insert_addin *Insert the model's answer*

Description

This function inserts a Codestral FIM into the current script.

Usage

```
insert_addin()
```

Value

`0` (invisible).

inventory_Rfiles *Inventory R files in the current directory*

Description

This function returns a data frame with the file names and paths of all R files in the current directory.

Usage

```
inventory_Rfiles()
```

Value

A data frame with the file names and paths of all R files in the current directory.

Examples

```
inventory_Rfiles()
```

`run_prompt` *Run a prompt from the console*

Description

This function allows the user to run a prompt directly from the console.

Usage

```
run_prompt(prompt = NULL)
```

Arguments

`prompt`	The prompt to run. If not provided, the user will be prompted to enter it.

Value

The result of the prompt.

Index

∗ **datasets**
ENDPOINTS, 7

allow_detect_package, 2

codestral, 3
codestral_init, 4
compile_dialog, 5
complete_current_script, 6

debug_mode, 6
detect_package, 7

ENDPOINTS, 7

include_file, 7
include_package_files, 8
insert_addin, 9
inventory_Rfiles, 9

run_prompt, 10