Package: perplexR (via r-universe)

August 27, 2024

Type Package

Title A Coding Assistant using Perplexity's Large Language Models

Version 0.0.3

Maintainer Gabriel Kaiser <quantresearch.gk@gmail.com>

Description A coding assistant using Perplexity's Large Language Models <https://www.perplexity.ai/> API. A set of functions and 'RStudio' add-ins that aim to help R developers.

License GPL (>= 3)

URL https://github.com/GabrielKaiserQFin/perplexR

BugReports https://github.com/GabrielKaiserQFin/perplexR/issues

Imports clipr, httr, jsonlite, miniUI, rstudioapi, shiny, utils

Encoding UTF-8

Language en-US

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

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

RemoteUrl https://github.com/gabrielkaiserqfin/perplexr

RemoteRef HEAD

RemoteSha ba4a3f005f37cf49e6e7a172a17242ecad87e3c0

Contents

`perplexR-package` *perplexR: A Coding Assistant using Perplexity's Large Language Models*

Description

A coding assistant using Perplexity's Large Language Models https://www.perplexity.ai/ API. A set of functions and 'RStudio' add-ins that aim to help R developers.

Author(s)

Maintainer: Gabriel Kaiser <quantresearch.gk@gmail.com>

See Also

Useful links:

- https://github.com/GabrielKaiserQFin/perplexR
- Report bugs at https://github.com/GabrielKaiserQFin/perplexR/issues

---

`annotateCode` *Large Language Model: Annotate code*

Description

Large Language Model: Annotate code

Usage

```
annotateCode(
  code = clipr::read_clip(allow_non_interactive = TRUE),
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant with extensive programming skills.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

code	The code to be commented by Large Language Model. If not provided, it will use what's copied on the clipboard.
PERPLEXITY_API_KEY	PERPLEXITY API key.
modelSelection	model choice. Default is mistral-7b-instruct.
systemRole	Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."
maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.
frequency_penalty	A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.
proxy	Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
annotateCode("z <- function(x) scale(x)^2")

## End(Not run)
```

`API_Request` *Get Large Language Model Completions Endpoint*

Description

Get Large Language Model Completions Endpoint

Usage

```
API_Request(
  prompt,
  PERPLEXITY_API_KEY = PERPLEXITY_API_KEY,
  modelSelection = modelSelection,
  systemRole = systemRole,
  maxTokens = maxTokens,
  temperature = temperature,
  top_p = top_p,
  top_k = top_k,
  presence_penalty = presence_penalty,
  frequency_penalty = frequency_penalty,
  proxy = proxy
)
```

Arguments

`prompt`	The prompt to generate completions for.
`PERPLEXITY_API_KEY`	PERPLEXITY API key.
`modelSelection`	model choice. Default is mistral-7b-instruct.
`systemRole`	Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."
`maxTokens`	The maximum integer of completion tokens returned by API.
`temperature`	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
`top_p`	Nucleus sampling threshold, valued between 0 and 1 inclusive.
`top_k`	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.

presence_penalty
: A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.

frequency_penalty
: A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.

proxy
: Default value is NULL.

AskMe *Ask Large Language Model*

Description

Note: See also clearChatSession.

Usage

```
AskMe(
  question,
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

question
: The question to ask Large Language Model.

PERPLEXITY_API_KEY
: PERPLEXITY API key.

modelSelection
: model choice. Default is mistral-7b-instruct.

systemRole
: Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."

maxTokens
: The maximum integer of completion tokens returned by API.

temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.
frequency_penalty	A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.
proxy	Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
AskMe("What do you think about Large language models?")

## End(Not run)
```

---

buildUnitTests *Large Language Model: Create Unit Tests*

Description

Create {testthat} test cases for the code.

Usage

```
buildUnitTests(
  code = clipr::read_clip(allow_non_interactive = TRUE),
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant with extensive programming skills.",
  maxTokens = 265,
  temperature = 1,
```

```
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

code	The code for which to create unit tests by Large Language Model. If not provided, it will use what's copied on the clipboard.
PERPLEXITY_API_KEY	PERPLEXITY API key.
modelSelection	model choice. Default is mistral-7b-instruct.
systemRole	Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."
maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.
frequency_penalty	A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.
proxy	Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
buildUnitTests("squared_numbers <- function(numbers) {\n  numbers ^ 2\n}")

## End(Not run)
```

clarifyCode *Large Language Model: Clarify Code*

Description

Large Language Model: Clarify Code

Usage

```
clarifyCode(
  code = clipr::read_clip(allow_non_interactive = TRUE),
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant with extensive programming skills.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

code	The code to be explained by Large Language Model. If not provided, it will use what's copied on the clipboard.
PERPLEXITY_API_KEY	PERPLEXITY API key.
modelSelection	model choice. Default is mistral-7b-instruct.
systemRole	Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."
maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.

frequency_penalty
: A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.

proxy
: Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
clarifyCode("z <- function(x) scale(x)^2")

## End(Not run)
```

---

debugCode *Large Language Model: Find Issues in Code*

Description

Large Language Model: Find Issues in Code

Usage

```
debugCode(
  code = clipr::read_clip(allow_non_interactive = TRUE),
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant with extensive programming skills.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

code	The code to be analyzed by Large Language Model. If not provided, it will use what's copied on the clipboard.
PERPLEXITY_API_KEY	PERPLEXITY API key.
modelSelection	model choice. Default is mistral-7b-instruct.
systemRole	Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."
maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.
frequency_penalty	A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.
proxy	Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
debugCode("z <- function(x) scale(x)2")

## End(Not run)
```

---

documentCode *Large Language Model: Code Documentation (roxygen2 style)*

Description

Large Language Model: Code Documentation (roxygen2 style)

Usage

```
documentCode(
  code = clipr::read_clip(allow_non_interactive = TRUE),
  inLineDocumentation = "roxygen2",
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant with extensive programming skills.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

code	The code to be documented by Large Language Model. If not provided, it will use what's copied on the clipboard.
inLineDocumentation	Formatting style of In-Line Documentation.
PERPLEXITY_API_KEY	PERPLEXITY API key.
modelSelection	model choice. Default is mistral-7b-instruct.
systemRole	Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."
maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.
frequency_penalty	A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.
proxy	Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
documentCode("z <- function(x) scale(x)^2")

## End(Not run)
```

`execAddin` *Run a Large Language Model as RStudio add-in*

Description

Run a Large Language Model as RStudio add-in

Usage

```
execAddin(FUN)
```

Arguments

`FUN`	The function to be executed.

`execAddin_AskMe` *Ask Large Language Model*

Description

Opens an interactive chat session with Large Language Model

Usage

```
execAddin_AskMe()
```

finishCode *Large Language Model: Finish code*

Description

Large Language Model: Finish code

Usage

```
finishCode(
  code = clipr::read_clip(allow_non_interactive = TRUE),
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant with extensive programming skills.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

code	The code to be completed by Large Language Model. If not provided, it will use what's copied on the clipboard.
PERPLEXITY_API_KEY	PERPLEXITY API key.
modelSelection	model choice. Default is mistral-7b-instruct.
systemRole	Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."
maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.

frequency_penalty
: A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.

proxy
: Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
finishCode("# A function to square each element of a vector\nsquare_each <- function(")

## End(Not run)
```

---

namingGenie *Large Language Model: Create a Function or Variable Name*

Description

Large Language Model: Create a Function or Variable Name

Usage

```
namingGenie(
  code = clipr::read_clip(allow_non_interactive = TRUE),
  namingConvention = "camelCase",
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant with extensive programming skills.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

code	The code for which to give a variable name to its result. If not provided, it will use what's copied on the clipboard.
namingConvention	Naming convention. Default is camelCase.
PERPLEXITY_API_KEY	PERPLEXITY API key.
modelSelection	model choice. Default is mistral-7b-instruct.
systemRole	Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."
maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.
frequency_penalty	A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.
proxy	Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
namingGenie("sapply(1:10, function(i) i ** 2)")

## End(Not run)
```

optimizeCode *Large Language Model: Optimize Code*

Description

Large Language Model: Optimize Code

Usage

```
optimizeCode(
  code = clipr::read_clip(allow_non_interactive = TRUE),
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant with extensive programming skills.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

code	The code to be optimized by Large Language Model. If not provided, it will use what's copied on the clipboard.
PERPLEXITY_API_KEY	PERPLEXITY API key.
modelSelection	model choice. Default is mistral-7b-instruct.
systemRole	Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."
maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.

`frequency_penalty`
: A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.

`proxy`
: Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
optimizeCode("z <- function(x) scale(x)^2")

## End(Not run)
```

`responseParser` *Parse Perplexity API Response*

Description

Takes the raw response from the Perplexity API and extracts the text content from it.

Usage

```
responseParser(raw)
```

Arguments

`raw`
: The raw object returned by the Perplexity API.

Value

Returns a character vector containing the text content of the response.

responseReturn *responseReturn*

Description

responseReturn

Usage

```
responseReturn(raw)
```

Arguments

raw	the chatresponse to return

Value

A character value with the response generated by Large Language Model.

rewriteText *Large Language Model: Rewrite Text*

Description

Large Language Model: Rewrite Text

Usage

```
rewriteText(
  text = clipr::read_clip(allow_non_interactive = TRUE),
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

text	The text to be rewritten by Large Language Model. If not provided, it will use what's copied on the clipboard.
PERPLEXITY_API_KEY	PERPLEXITY API key.
modelSelection	model choice. Default is mistral-7b-instruct.
systemRole	Role for model. Default is: "You are a helpful assistant."
maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.
frequency_penalty	A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.
proxy	Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
rewriteText("Dear Recipient, I hope this message finds you well.")

## End(Not run)
```

---

translateCode *Translate Code from One Language to Another*

Description

This function takes a snippet of code and translates it from one programming language to another using Perplexity API. The default behavior is to read the code from the clipboard and translate from R to Python.

Usage

```
translateCode(
  code = clipr::read_clip(allow_non_interactive = TRUE),
  from = "R",
  to = "Python",
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant with extensive programming skills.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

code	A string containing the code to be translated. If not provided, the function will attempt to read from the clipboard.
from	The language of the input code. Default is "R".
to	The target language for translation. Default is "Python".
PERPLEXITY_API_KEY	PERPLEXITY API key.
modelSelection	model choice. Default is mistral-7b-instruct.
systemRole	Role for model. Default is: "You are a helpful assistant with extensive knowledge of R programming."
maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.
frequency_penalty	A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.
proxy	Default value is NULL.

Value

A string containing the translated code.

Examples

```
## Not run:
translateCode("your R code here", from = "R", to = "Python")

## End(Not run)
```

`translateText` *Large Language Model: Translate Text*

Description

Large Language Model: Translate Text

Usage

```
translateText(
  text = clipr::read_clip(allow_non_interactive = TRUE),
  toLanguage = "German",
  PERPLEXITY_API_KEY = Sys.getenv("PERPLEXITY_API_KEY"),
  modelSelection = c("mistral-7b-instruct", "mixtral-8x7b-instruct",
    "codellama-70b-instruct", "sonar-small-chat", "sonar-small-online",
    "sonar-medium-chat", "sonar-medium-online"),
  systemRole = "You are a helpful assistant.",
  maxTokens = 265,
  temperature = 1,
  top_p = NULL,
  top_k = 100,
  presence_penalty = 0,
  frequency_penalty = NULL,
  proxy = NULL
)
```

Arguments

`text`	The text to be translated by Large Language Model. If not provided, it will use what's copied on the clipboard.
`toLanguage`	The language to be translated into.
`PERPLEXITY_API_KEY`	PERPLEXITY API key.
`modelSelection`	model choice. Default is mistral-7b-instruct.
`systemRole`	Role for model. Default is: "You are a helpful assistant."

maxTokens	The maximum integer of completion tokens returned by API.
temperature	The amount of randomness in the response, valued between 0 inclusive and 2 exclusive. Higher values are more random, and lower values are more deterministic. Set either temperature or top_p.
top_p	Nucleus sampling threshold, valued between 0 and 1 inclusive.
top_k	The number of tokens to keep for highest top-k filtering, specified as an integer between 0 and 2048 inclusive. If set to 0, top-k filtering is disabled.
presence_penalty	A value between -2.0 and 2.0. Positive values penalize new tokens based on whether they appear in the text so far, increasing the model's likelihood to talk about new topics. Incompatible with frequency_penalty.
frequency_penalty	A multiplicative penalty greater than 0. Values greater than 1.0 penalize new tokens based on their existing frequency in the text so far, decreasing the model's likelihood to repeat the same line verbatim. A value of 1.0 means no penalty.
proxy	Default value is NULL.

Value

A character value with the response generated by Large Language Model.

Examples

```
## Not run:
translateText("Dear Recipient, I hope this message finds you well.")

## End(Not run)
```

Index