

Package: rgemini (via r-universe)

February 21, 2025

Type Package
Title R interface to Google Gemini API
Version 0.1.0
Author Sebastian Kranz
Maintainer Sebastian Kranz <sebastian.kranz@uni-ulm.de>
Description In development
License GPL>=2.0
Depends jsonlite, httr
Encoding UTF-8
LazyData true
RoxygenNote 7.3.1
Config/pak/sysreqs libssl-dev
Repository <https://skranz.r-universe.dev>
RemoteUrl <https://github.com/skranz/rgemini>
RemoteRef main
RemoteSha 00874bfd917352fe340fd0f641b7570057453b61

Contents

arr_resp	2
gemini_content	2
gemini_media_upload	3
obj_resp	3
response_schema	4
run_gemini	4
set_gemini_api_key	6

Index	7
--------------	----------

arr_resp	<i>Create an Array Response Template</i>
----------	--

Description

This function creates a template for an array response consisting of JSON objects. Each element (object) in the array is expected to have the provided fields.

Usage

```
arr_resp(...)
```

Arguments

... Named elements that define the fields for each JSON object in the array.

Value

A list with class 'arr_resp' that serves as a template for an array response.

Examples

```
# Create an array response template with two fields.  
arr_template <- arr_resp(city = "Paris", country = "France")  
str(arr_template)
```

gemini_content	<i>Extract content from a more detailed run_gemini response</i>
----------------	---

Description

Extract content from a more detailed run_gemini response

Usage

```
gemini_content(result)
```

gemini_media_upload	<i>Upload a document or image to Gemini API.</i>
---------------------	--

Description

Returned object or a list of such objects can then be passed to `gemini_run`. Media should be available for around an hour.

Usage

```
gemini_media_upload(  
  file_path,  
  mime_type = NULL,  
  display_name = NULL,  
  api_key = getOption("gemini_api_key"),  
  verbose = FALSE  
)
```

obj_resp	<i>Create an Object Response Template</i>
----------	---

Description

This function creates a template for a single JSON object response. The response is expected to be a JSON object with the provided fields.

Usage

```
obj_resp(...)
```

Arguments

... Named elements that define the fields for the JSON object.

Value

A list with class 'obj_resp' that serves as a template for an object response.

Examples

```
# Create an object response template with three fields.  
obj_template <- obj_resp(city = "Paris", country = "France", population = 5.2)  
str(obj_template)
```

response_schema	<i>Generate a JSON Schema from a Response Template</i>
-----------------	--

Description

This function recursively generates a JSON Schema based on a response template. The template must be created using either `obj_resp` (for a single object response) or `arr_resp` (for an array response). The function infers the JSON Schema by inspecting the data types of the provided fields without including the example values.

Usage

```
response_schema(example)
```

Arguments

example	A response template created by <code>obj_resp</code> or <code>arr_resp</code> .
---------	---

Value

A list representing the inferred JSON Schema for the provided response template.

run_gemini	<i>Generate Content with Gemini API</i>
------------	---

Description

Sends a text prompt (and optionally one or more media objects) to the Gemini API to generate content.

Usage

```
run_gemini(  
  prompt,  
  model = "gemini-2.0-flash",  
  media = NULL,  
  json_mode = !is.null(response_schema),  
  response_schema = NULL,  
  temperature = 0.1,  
  add_prompt = FALSE,  
  verbose = FALSE,  
  api_key = getOption("gemini_api_key"),  
  just_content = TRUE,  
  httr_response = FALSE  
)
```

Arguments

prompt	A character string containing the text prompt to be sent to the Gemini API.
model	A character string specifying the Gemini model to use. Defaults to "gemini-2.0-flash".
media	Either a single media object or a list of media objects to be included in the prompt. Call gemini_media_upload to upload the document or media.
json_mode	Logical. If TRUE, expects the response in JSON format. Defaults to !is.null(response_schema).
response_schema	Optional structured output schema for the response. Defaults to NULL.
temperature	A numeric value controlling the randomness of the output. Defaults to 0.1.
add_prompt	Logical. If TRUE, includes the prompt in the returned result. Defaults to FALSE.
verbose	Logical. If TRUE, prints debugging and request information. Defaults to FALSE.
api_key	A character string containing your Gemini API key. Defaults to the value obtained from <code>getOption("gemini_api_key")</code> .
just_content	Logical. If TRUE, returns only the content portion of the result. JSON is transformed to R. Defaults to TRUE. Otherwise a data frame with the response of the POST call is returned, which includes fields like <code>status_code</code> .
httr_response	Logical. Only relevant if <code>just_content=FALSE</code> . Returns POST response in original format, not transformed to a more convenient data set.

Details

The function builds a JSON payload that includes the provided text prompt as well as any media objects supplied via the `media` parameter. Each media object is appended to the `parts` of the request under the `file_data` key. The payload is then sent via a POST request to the Gemini API endpoint for content generation.

Value

A list containing the Gemini API response. If `as_data_frame` is TRUE, the response is converted to a data frame.

See Also

[POST](#), [toJSON](#)

Examples

```
## Not run:
# Example using only a text prompt:
result <- run_gemini(prompt = "Tell a joke", verbose = TRUE)

## End(Not run)
```

set_gemini_api_key *Set your Gemini API*

Description

Set your Gemini API

Usage

```
set_gemini_api_key(key = NULL, file = NULL)
```

Index

`arr_resp`, [2](#)

`gemini_content`, [2](#)

`gemini_media_upload`, [3](#), [5](#)

`obj_resp`, [3](#)

POST, [5](#)

`response_schema`, [4](#)

`run_gemini`, [4](#)

`set_gemini_api_key`, [6](#)

`toJSON`, [5](#)