# Package 'gemini.R'

April 10, 2025

**Title** Interface for 'Google Gemini' API

**Version** 0.12.0

**Maintainer** Jinhwan Kim <hwanistic@gmail.com>

**Description** Provides a comprehensive interface for Google Gemini API,
enabling users to access and utilize Gemini Large Language Model (LLM) functionalities directly from R.
This package facilitates seamless integration with Google Gemini, allowing for advanced language processing,
text generation, and other AI-driven capabilities within the R environment.
For more information, please visit <https://ai.google.dev/docs/gemini_api_overview>.

**License** MIT + file LICENSE

**Depends** R (>= 4.1.0)

**URL** https://github.com/jhk0530/gemini.R

**BugReports** https://github.com/jhk0530/gemini.R/issues

**Encoding** UTF-8

**Imports** base64enc, cli, httr2, jsonlite, rstudioapi, tools

**RoxygenNote** 7.3.2

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Config/Needs/website** rmarkdown

**NeedsCompilation** no

**Author** Jinhwan Kim [aut, cre, cph] (<https://orcid.org/0009-0009-3217-2417>),
Maciej Nasinski [ctb]

**Repository** CRAN

**Date/Publication** 2025-04-10 10:10:02 UTC

# Contents



---

`addHistory` *Add history for chating context*

---

## Description

Add history for chating context

## Usage

```
addHistory(history, role = NULL, item = NULL)
```

## Arguments

| | |
|---|---|
| `history` | The history of chat |
| `role` | The role of chat: "user" or "model" |
| `item` | The item of chat: "prompt" or "output" |

## Value

The history of chat

countTokens *Count Tokens for Gemini Content (Including Images)*

## Description

Calculates the token count for a given content, including text and image data, using the Vertex AI Gemini API.

## Usage

```
countTokens(
  jsonkey = NULL,
  model_id = NULL,
  content = NULL,
  region = "us-central1"
)
```

## Arguments

| | |
|---|---|
| jsonkey | A path to JSON file containing the service account key from Vertex AI. |
| model_id | The ID of the Gemini model. |
| content | The content (text, image, or list of text/image parts) for which to count tokens.<br>• For text, provide a string.<br>• For images, provide a list with data (base64 encoded image) and mimeType (e.g., "image/png", "image/jpeg").<br>• For multiple content parts, provide a list where each element is either a text string or an image list. |
| region | The Google Cloud region where your Vertex AI resources are located (default is "us-central1"). See https://cloud.google.com/vertex-ai/docs/regions for available regions. |

## Value

A numeric value representing the token count of the content.

## Examples

```
## Not run:
library(gemini.R)

# For text content
key_file <- "YOURAPIKEY.json"
model <- "2.0-flash"
token_count_text <- countTokens(
  jsonkey = key_file,
  model_id = model,
  content = "Hello, world!"
```

```
)
print(token_count_text)

# For image content (assuming 'image.jpg' is in your working directory)
image_data <- base64enc::base64encode("image.jpg")
image_content <- list(data = image_data, mimeType = "image/jpeg")
token_count_image <- countTokens(
  jsonkey = key_file,
  model_id = model,
  content = image_content
)
print(token_count_image)

# For multiple content parts (text and image)
content_parts <- list(
  list(text = "This is the first part."),
  list(data = image_data, mimeType = "image/jpeg"),
  list(text = "This is the last part")
)
token_count_parts <- countTokens(
  jsonkey = key_file,
  model_id = model,
  content = content_parts
)
print(token_count_parts)

## End(Not run)
```

---

gemini *Generate text from text with Gemini*

---

## Description

Generate text from text with Gemini

## Usage

```
gemini(
  prompt,
  model = "2.0-flash",
  temperature = 1,
  maxOutputTokens = 8192,
  topK = 40,
  topP = 0.95,
  seed = 1234
)
```

### Arguments

| | |
|---|---|
| prompt | The prompt to generate text from |
| model | The model to use. Options are "2.0-flash", "2.0-flash-lite", "2.5-pro-exp-03-25". Default is '2.0-flash'. see https://ai.google.dev/gemini-api/docs/models/gemini |
| temperature | The temperature to use. Default is 1 value should be between 0 and 2 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| maxOutputTokens | The maximum number of tokens to generate. Default is 8192 and 100 tokens correspond to roughly 60-80 words. |
| topK | The top-k value to use. Default is 40 value should be between 0 and 100 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| topP | The top-p value to use. Default is 0.95 value should be between 0 and 1 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| seed | The seed to use. Default is 1234 value should be integer see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |

### Value

Generated text

### See Also

https://ai.google.dev/docs/gemini_api_overview#text_input

### Examples

```
## Not run:
library(gemini.R)
setAPI("YOUR_API_KEY")
gemini("Explain dplyr's mutate function")

## End(Not run)
```

---

## gemini.vertex *Generate text from text with Gemini Vertex API*

---

### Description

Generate text from text with Gemini Vertex API

## Usage

```
gemini.vertex(
  prompt = NULL,
  tokens = NULL,
  temperature = 1,
  maxOutputTokens = 8192,
  topK = 40,
  topP = 0.95,
  seed = 1234
)
```

## Arguments

| | |
|---|---|
| prompt | A character string containing the prompt for the Gemini model. |
| tokens | A list containing the API URL and key from token.vertex() function. |
| temperature | The temperature to use. Default is 1 value should be between 0 and 2 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| maxOutputTokens | The maximum number of tokens to generate. Default is 8192 and 100 tokens correspond to roughly 60-80 words. |
| topK | The top-k value to use. Default is 40 value should be between 0 and 100 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| topP | The top-p value to use. Default is 0.95 value should be between 0 and 1 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| seed | The seed to use. Default is 1234 value should be integer see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |

## Value

A character string containing the generated text.

## See Also

https://ai.google.dev/docs/gemini_api_overview#text_input

## Examples

```
## Not run:
# token should be created before this. using the token.vertex() function
prompt <- "What is sachins Jersey number?"
gemini.vertex(prompt, tokens)

## End(Not run)
```

gemini_audio *Analyze audio using Gemini*

## Description

This function sends audio to the Gemini API and returns a text description.

## Usage

```
gemini_audio(
  audio = NULL,
  prompt = "Describe this audio",
  model = "2.0-flash",
  temperature = 1,
  maxOutputTokens = 8192,
  topK = 40,
  topP = 0.95,
  seed = 1234
)
```

## Arguments

| | |
|---|---|
| audio | Path to the audio file (default: uses a sample file). Must be an MP3. |
| prompt | A string describing what to do with the audio. |
| model | The model to use. Options are "2.0-flash", "2.0-flash-lite", "2.5-pro-exp-03-25". Default is '2.0-flash' see https://ai.google.dev/gemini-api/docs/models/gemini |
| temperature | The temperature to use. Default is 1 value should be between 0 and 2 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| maxOutputTokens | The maximum number of tokens to generate. Default is 8192 and 100 tokens correspond to roughly 60-80 words. |
| topK | The top-k value to use. Default is 40 value should be between 0 and 100 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| topP | The top-p value to use. Default is 0.95 value should be between 0 and 1 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| seed | The seed to use. Default is 1234 value should be integer see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |

## Value

A character vector containing the Gemini API's response.

## Examples

```
## Not run:
library(gemini.R)
setAPI("YOUR_API_KEY")
gemini_audio(audio = system.file("docs/reference/helloworld.mp3", package = "gemini.R"))

## End(Not run)
```

---

`gemini_audio.vertex` *Analyze Audio using Gemini Vertex API*

---

## Description

This function sends audio to the Gemini API and returns a text description.

## Usage

```
gemini_audio.vertex(
  audio = NULL,
  prompt = "Describe this audio",
  tokens = NULL,
  temperature = 1,
  maxOutputTokens = 8192,
  topK = 40,
  topP = 0.95,
  seed = 1234
)
```

## Arguments

| | |
|---|---|
| `audio` | Path to the audio file (character string). only supports "mp3". |
| `prompt` | A prompt to guide the Gemini API's analysis (character string, defaults to "Describe this audio"). |
| `tokens` | A list containing the API URL and key from token.vertex() function. |
| `temperature` | The temperature to use. Default is 1 value should be between 0 and 2 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| `maxOutputTokens` | The maximum number of tokens to generate. Default is 8192 and 100 tokens correspond to roughly 60-80 words. |
| `topK` | The top-k value to use. Default is 40 value should be between 0 and 100 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| `topP` | The top-p value to use. Default is 0.95 value should be between 0 and 1 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| `seed` | The seed to use. Default is 1234 value should be integer see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |

## Value

A character vector containing the Gemini API's description of the audio.

---

`gemini_chat` *Multi-turn conversations (chat)*

---

## Description

Generate text from text with Gemini

## Usage

```
gemini_chat(
  prompt,
  history = list(),
  model = "2.0-flash",
  temperature = 1,
  maxOutputTokens = 8192,
  topK = 40,
  topP = 0.95,
  seed = 1234
)
```

## Arguments

| | |
|---|---|
| `prompt` | The prompt to generate text from |
| `history` | history object to keep track of the conversation |
| `model` | The model to use. Options are "2.0-flash", "2.0-flash-lite", "2.5-pro-exp-03-25". Default is '2.0-flash' see https://ai.google.dev/gemini-api/docs/models/gemini |
| `temperature` | The temperature to use. Default is 1 value should be between 0 and 2 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| `maxOutputTokens` | The maximum number of tokens to generate. Default is 8192 and 100 tokens correspond to roughly 60-80 words. |
| `topK` | The top-k value to use. Default is 40 value should be between 0 and 100 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| `topP` | The top-p value to use. Default is 0.95 value should be between 0 and 1 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| `seed` | The seed to use. Default is 1234 value should be integer see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |

## Value

Generated text

## See Also

https://ai.google.dev/docs/gemini_api_overview#chat

## Examples

```
## Not run:
library(gemini.R)
setAPI("YOUR_API_KEY")

chats <- gemini_chat("Pretend you're a snowman and stay in character for each")
print(chats$outputs)

chats <- gemini_chat("What's your favorite season of the year?", chats$history)
print(chats$outputs)

chats <- gemini_chat("How do you think about summer?", chats$history)
print(chats$outputs)

## End(Not run)
```

---

# gemini_image *Generate text from text and image with Gemini*

---

## Description

Generate text from text and image with Gemini

## Usage

```
gemini_image(
  image = NULL,
  prompt = "Explain this image",
  model = "2.0-flash",
  temperature = 1,
  maxOutputTokens = 8192,
  topK = 40,
  topP = 0.95,
  seed = 1234,
  type = "png"
)
```

## Arguments

| | |
|---|---|
| image | The image to generate text |
| prompt | The prompt to generate text, Default is "Explain this image" |
| model | The model to use. Options are "2.0-flash", "2.0-flash-lite", "2.5-pro-exp-03-25". Default is '2.0-flash' see https://ai.google.dev/gemini-api/docs/models/gemini |

temperature — The temperature to use. Default is 1 value should be between 0 and 2 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters

maxOutputTokens — The maximum number of tokens to generate. Default is 8192 and 100 tokens correspond to roughly 60-80 words.

topK — The top-k value to use. Default is 40 value should be between 0 and 100 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters

topP — The top-p value to use. Default is 0.95 value should be between 0 and 1 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters

seed — The seed to use. Default is 1234 value should be integer see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters

type — The type of image. Options are 'png', 'jpeg', 'webp', 'heic', 'heif'. Default is 'png'

## Value

Generated text

## See Also

https://ai.google.dev/docs/gemini_api_overview#text_image_input

## Examples

```
## Not run:
library(gemini.R)
setAPI("YOUR_API_KEY")
gemini_image(image = system.file("docs/reference/figures/image.png", package = "gemini.R"))

## End(Not run)
```

---

# gemini_image.vertex *Generate text from text and image with Gemini Vertex API*

---

## Description

Generate text from text and image with Gemini Vertex API

## Usage

```
gemini_image.vertex(
  image = NULL,
  prompt = "Explain this image",
  type = "png",
  tokens = NULL,
  temperature = 1,
```

```
  maxOutputTokens = 8192,
  topK = 40,
  topP = 0.95,
  seed = 1234
)
```

## Arguments

| | |
|---|---|
| image | The image to generate text |
| prompt | A character string specifying the prompt to use with the image. Defaults to "Explain this image". |
| type | A character string specifying the image type ("png", "jpeg", "webp", "heic", "heif"). Defaults to "png". |
| tokens | A list containing the API URL and key from token.vertex() function. |
| temperature | The temperature to use. Default is 1 value should be between 0 and 2 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| maxOutputTokens | The maximum number of tokens to generate. Default is 8192 and 100 tokens correspond to roughly 60-80 words. |
| topK | The top-k value to use. Default is 40 value should be between 0 and 100 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| topP | The top-p value to use. Default is 0.95 value should be between 0 and 1 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| seed | The seed to use. Default is 1234 value should be integer see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |

## Value

A character string containing Gemini's description of the image.

---

# gemini_search *Generate text with real-time information using Google Search (Grounding)*

---

## Description

Generate text responses that include up-to-date information from Google Search

## Usage

```
gemini_search(
  prompt,
  temperature = 1,
  maxOutputTokens = 8192,
  topK = 40,
  topP = 0.95,
  seed = 1234
)
```

### Arguments

| | |
|---|---|
| prompt | The prompt or question requiring real-time information |
| temperature | The temperature to use. Default is 1 value should be between 0 and 2 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| maxOutputTokens | The maximum number of tokens to generate. Default is 8192 and 100 tokens correspond to roughly 60-80 words. |
| topK | The top-k value to use. Default is 40 value should be between 0 and 100 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| topP | The top-p value to use. Default is 0.95 value should be between 0 and 1 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| seed | The seed to use. Default is 1234 value should be integer see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |

### Value

Generated text with real-time information from Google Search

### See Also

https://ai.google.dev/docs/search_retrieval

### Examples

```
## Not run:
library(gemini.R)
setAPI("YOUR_API_KEY")
gemini_search("What is the current Google stock price?")

## End(Not run)
```

---

## gemini_searchR *Generate text with real-time information using Gemini (Retrieval)*

---

### Description

Generate text responses with simplified access to Gemini models

### Usage

```
gemini_searchR(
  prompt,
  model = "1.5-flash",
  temperature = 1,
  maxOutputTokens = 8192,
  topK = 40,
```

```
  topP = 0.95,
  seed = 1234
)
```

## Arguments

| | |
|---|---|
| prompt | The prompt or question to ask |
| model | The model to use. Options are "1.5-flash", "1.5-pro". Default is '1.5-flash'. see https://ai.google.dev/gemini-api/docs/models/gemini |
| temperature | The temperature to use. Default is 1 value should be between 0 and 2 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| maxOutputTokens | The maximum number of tokens to generate. Default is 8192 and 100 tokens correspond to roughly 60-80 words. |
| topK | The top-k value to use. Default is 40 value should be between 0 and 100 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| topP | The top-p value to use. Default is 0.95 value should be between 0 and 1 see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |
| seed | The seed to use. Default is 1234 value should be integer see https://ai.google.dev/gemini-api/docs/models/generative-models#model-parameters |

## Value

Generated text response from the Gemini model

## See Also

https://ai.google.dev/docs/gemini_api_overview#text_input

## Examples

```
## Not run:
library(gemini.R)
setAPI("YOUR_API_KEY")
gemini_searchR("Who won the latest F1 grand prix?")

## End(Not run)
```

---

# gen_docs *Generate Roxygen Documentation*

---

## Description

Generates Roxygen2 documentation for an R function based on the currently selected code.

### Usage

```
gen_docs(prompt = NULL)
```

### Arguments

| | |
|---|---|
| prompt | A character string specifying additional instructions for the LLM. Defaults to a prompt requesting Roxygen2 documentation without the original code. |

### Value

A character string containing the generated Roxygen2 documentation.

---

## gen_image *Generate and save image using Gemini*

---

### Description

Generate an image using Gemini's image generation capabilities and save it to a file

### Usage

```
gen_image(
  prompt,
  filename = "gemini_image.png",
  overwrite = TRUE,
  model = "2.0-flash-exp-image-generation",
  temperature = 1,
  seed = 1234
)
```

### Arguments

| | |
|---|---|
| prompt | The prompt to generate an image from |
| filename | The filename to save the image to. Default is "gemini_image.png" |
| overwrite | Logical, whether to overwrite existing file. Default is TRUE |
| model | The model to use. Default is "2.0-flash-exp-image-generation" |
| temperature | The temperature to use. Default is 1 |
| seed | The seed to use. Default is 1234 |

### Value

The path to the saved file

## Examples

```
## Not run:
library(gemini.R)
setAPI("YOUR_API_KEY")
gen_image("Create an image of a cat wearing sunglasses")

## End(Not run)
```

---

`gen_tests` *Generates unit test code for an R function.*

---

## Description

Generates unit test code for an R function.

## Usage

```
gen_tests(prompt = NULL)
```

## Arguments

| | |
|---|---|
| `prompt` | A character string specifying the prompt for the Gemini model. If NULL, a default prompt is used. |

## Value

#' A character string containing the generated unit test code.

---

`setEnv` *Store API key in local environment file*

---

## Description

Saves the API key to a local .Renviron file for persistent access across R sessions

## Usage

```
setEnv(api_key, overwrite = TRUE, install_message = TRUE)
```

## Arguments

| | |
|---|---|
| `api_key` | The API key to store |
| `overwrite` | Whether to overwrite the existing API key if already present in .Renviron (default: TRUE) |
| `install_message` | Whether to display a message about how to use the API (default: TRUE) |

## Value

No return value, called for side effects.

## See Also

setAPI which sets the API key for the current session only

## Examples

```
## Not run:
setEnv("your_api_key")

## End(Not run)
```

---

# token.vertex *Generate Gemini Access Token and Endpoint URL*

---

## Description

Generates an access token for the Gemini model and constructs the corresponding endpoint URL.

## Usage

```
token.vertex(
  jsonkey = NULL,
  model_id = NULL,
  expTime = 3600,
  region = "us-central1"
)
```

## Arguments

| | |
|---|---|
| jsonkey | A path to JSON file containing the service account key from Vertex AI. |
| model_id | The ID of the Gemini model. This will be prepended with "gemini-". |
| expTime | The expiration time of the access token in seconds (default is 3600 seconds, or 1 hour). |
| region | The Google Cloud region where your Vertex AI resources are located (default is "us-central1"). See https://cloud.google.com/vertex-ai/docs/general/locations for available regions. |

## Value

A list containing:

| | |
|---|---|
| key | The generated access token. |
| url | The endpoint URL for the Gemini model. |

## Examples

```
## Not run:
library(gemini.R)
tokens <- token.vertex(jsonkey = "YOURAPIKEY.json", model_id = "1.5-flash")

# Specify a different region
tokens <- token.vertex(jsonkey = "YOURAPIKEY.json", model_id = "1.5-flash", region = "europe-west4")

## End(Not run)
```

# Index