

# Package ‘rosetteApi’

February 12, 2019

**Type** Package

**Title** 'Rosette' API

**Version** 1.12.1

**Date** 2019-02-12

**Description** 'Rosette' is an API for multilingual text analysis and information extraction. More information can be found at <<https://developer.rosette.com>>.

**License** Apache License 2.0 | file LICENSE

**Depends** R (>= 3.1.0)

**Imports** httr,  
jsonlite

**Suggests** knitr,  
rmarkdown,  
testthat

**URL** <https://developer.rosette.com>

**BugReports** <https://developer.rosette.com/support-ticket>

**VignetteBuilder** knitr

**RoxygenNote** 6.0.1

## R topics documented:

api . . . . .	2
check_content_parameters . . . . .	3
check_deduplication . . . . .	3
check_for_multipart . . . . .	4
check_morphology . . . . .	4
check_names . . . . .	5
create_multipart . . . . .	5
error_check . . . . .	6
get_binding_version . . . . .	6
get_endpoint . . . . .	7
get_headers . . . . .	7

post_endpoint . . . . .	8
serialize_name_deduplication_parameters . . . . .	8
serialize_name_parameters . . . . .	9
serialize_parameters . . . . .	9
<b>Index</b>	<b>10</b>

---

api	<i>api wrapper function that checks for a user_key and determines the Rosette API endpoint to be utilized</i>
-----	---

---

**Description**

api wrapper function that checks for a user\_key and determines the Rosette API endpoint to be utilized

**Usage**

```
api(user_key, endpoint, parameters = FALSE, custom_headers = NULL,
    url_parameters = NULL, url = "https://api.rosette.com/rest/v1/")
```

**Arguments**

- user\_key - Rosette API authentication key
- endpoint - Rosette API endpoint to be utilized
- parameters - parameters list to be passed to specified Rosette API endpoint
- custom\_headers - custom headers for Rosette Api
- url\_parameters - query parameters
- url - url for Rosette Api

**Value**

Returns a list(content, header)

**Examples**

```
## Not run:
parameters <- list()
parameters[["content"]] <- "Bill Murray will appear in new Ghostbusters film."
response = api(01234567890, "entities", parameters) will return response$content and response$header

## End(Not run)
```

---

`check_content_parameters`*preemptive check for content/contentUri request parameter errors*

---

**Description**

preemptive check for content/contentUri request parameter errors

**Usage**

`check_content_parameters(parameters)`

**Arguments**

`parameters`      - parameters list to be passed to specified Rosette API endpoint

**Value**

Returns list of verified parameters to be sent to Rosette API

---

`check_deduplication`      *check if the required request parameters for name deduplication are correct*

---

**Description**

check if the required request parameters for name deduplication are correct

**Usage**

`check_deduplication(parameters, endpoint)`

**Arguments**

`parameters`      - parameters list to be passed to specified Rosette API endpoint  
`endpoint`          - Rosette API endpoint to be utilized

**Value**

Returns list of verified parameters to be sent to Rosette API

---

check_for_multipart	<i>Helper to check for file submission</i>
---------------------	--

---

**Description**

Helper to check for file submission

**Usage**

check\_for\_multipart(parameters)

**Arguments**

parameters      - JSON parameters

**Value**

true if multipart

---

check_morphology	<i>determine which morphology endpoint to use</i>
------------------	---

---

**Description**

determine which morphology endpoint to use

**Usage**

check\_morphology(parameters, endpoint)

**Arguments**

parameters      - parameters list to be passed to specified Rosette API endpoint  
endpoint        - Rosette API endpoint to be utilized

**Value**

Returns the specified morphology function

---

check_names	<i>check if the required request parameters for either names endpoint are correct</i>
-------------	---

---

**Description**

check if the required request parameters for either names endpoint are correct

**Usage**

```
check_names(parameters, endpoint)
```

**Arguments**

parameters	- parameters list to be passed to specified Rosette API endpoint
endpoint	- Rosette API endpoint to be utilized

**Value**

Returns list of verified parameters to be sent to Rosette API

---

create_multipart	<i>create a multipart</i>
------------------	---------------------------

---

**Description**

create a multipart

**Usage**

```
create_multipart(parameters)
```

**Arguments**

parameters	- parameters list to be passed to specified Rosette API endpoint
------------	--

**Value**

Returns a multipart

---

error_check	<i>check if Rosette API response includes and error message</i>
-------------	---

---

**Description**

check if Rosette API response includes and error message

**Usage**

```
error_check(response)
```

**Arguments**

response      - response from Rosette API

**Value**

Returns an error if one exists or the Rosette API response

---

get_binding_version	<i>Provides the binding Version</i>
---------------------	-------------------------------------

---

**Description**

Provides the binding Version

**Usage**

```
get_binding_version()
```

**Value**

current binding version

---

get_endpoint	<i>http::GET request to specified Rosette API endpoint</i>
--------------	--

---

**Description**

http::GET request to specified Rosette API endpoint

**Usage**

```
get_endpoint(user_key, endpoint, url, custom_headers = NULL,  
             url_parameters = NULL)
```

**Arguments**

user_key	- Rosette API authentication key
endpoint	- Rosette API endpoint to be utilized
url	- url for Rosette Api
custom_headers	- custom headers for Rosette Api
url_parameters	- url query parameters

**Value**

Returns the Rosette API response

---

get_headers	<i>Helper to combine custom headers with the default ones</i>
-------------	---

---

**Description**

Helper to combine custom headers with the default ones

**Usage**

```
get_headers(user_key, content_type = "application/json",  
            custom_headers = NULL)
```

**Arguments**

user_key	- Rosette API authentication key
content_type	- Header Content-Type
custom_headers	- custom headers for the Rosette API

**Value**

Returns the combined h

---

post_endpoint	<i>http::POST request to specified Rosette API endpoint</i>
---------------	---

---

**Description**

http::POST request to specified Rosette API endpoint

**Usage**

```
post_endpoint(user_key, parameters, endpoint, url, custom_headers = NULL,  
              url_parameters = NULL)
```

**Arguments**

user_key	- Rosette API authentication key
parameters	- parameters list to be passed to specified Rosette API endpoint
endpoint	- Rosette API endpoint to be utilized
url	- url for Rosette Api
custom_headers	- custom headers for Rosette Api
url_parameters	- query parameters

**Value**

Returns the Rosette API response

---

serialize_name_deduplication_parameters	<i>serialize Rosette API parameters</i>
---	---

---

**Description**

serialize Rosette API parameters

**Usage**

```
serialize_name_deduplication_parameters(parameters)
```

**Arguments**

parameters	- parameters list to be passed to name-deduplication
------------	--

**Value**

Returns the serialized parameters for the Rosette API



---

serialize\_name\_parameters  
*serialize Rosette API parameters*

---

**Description**

serialize Rosette API parameters

**Usage**

serialize\_name\_parameters(parameters)

**Arguments**

parameters      - parameters list to be passed to either name-translation or name-similarity

**Value**

Returns the serialized parameters for the Rosette API

---

serialize\_parameters      *serialize Rosette API parameters*

---

**Description**

serialize Rosette API parameters

**Usage**

serialize\_parameters(parameters)

**Arguments**

parameters      - parameters list to be passed to specified Rosette API endpoint

**Value**

Returns the serialized parameters for the Rosette API

# Index

api, [2](#)

check\_content\_parameters, [3](#)

check\_deduplication, [3](#)

check\_for\_multipart, [4](#)

check\_morphology, [4](#)

check\_names, [5](#)

create\_multipart, [5](#)

error\_check, [6](#)

get\_binding\_version, [6](#)

get\_endpoint, [7](#)

get\_headers, [7](#)

post\_endpoint, [8](#)

serialize\_name\_deduplication\_parameters,  
[8](#)

serialize\_name\_parameters, [9](#)

serialize\_parameters, [9](#)