

Package: cmmr (via r-universe)

August 22, 2024

Type Package

Title CEU Mass Mediator RESTful API

Version 1.0.1

Date 2020-08-27

Depends R (>= 3.1.0)

Imports httr (>= 1.3.1), progress (>= 1.2.0), RJSONIO (>= 1.3-0)

Author Yaoxiang Li <yl814@georgetown.edu>, Alberto Gil de la Fuente<alberto.gilf@gmail.com>, Charles P. Hinzman <cph51@georgetown.edu>, Amrita K. Cheema <akc27@georgetown.edu>

Maintainer Yaoxiang Li <yl814@georgetown.edu>

Description CEU (CEU San Pablo University) Mass Mediator is an on-line tool for aiding researchers in performing metabolite annotation. 'cmmr' (CEU Mass Mediator RESTful API) allows for programmatic access in R: batch search, batch advanced search, MS/MS (tandem mass spectrometry) search, etc. For more information about the API Endpoint please go to <<https://github.com/lzyacht/cmmr>>.

License GPL-3

Encoding UTF-8

LazyData true

URL <https://github.com/lzyacht/cmmr>

RoxygenNote 6.1.1

Suggests testthat

Repository <https://yaoxiangli.r-universe.dev>

RemoteUrl <https://github.com/yaoxiangli/cmmr>

RemoteRef HEAD

RemoteSha 9b4b71e534a363b271fb720edfd15fe40461f664

Contents

advanced_batch_search	2
batch_search	4
create_advanced_batch_body	5
create_batch_body	6
create_msms_body	7
msms_search	8

Index	10
--------------	-----------

advanced_batch_search *Encapsulation of CEU Mass Mediator advanced batch search API.*

Description

advanced_batch_search returns the body string of a POST request. using the following code to install the dependencies: `install.packages(c("httr", "progress", "RJSONIO"))`

Usage

```
advanced_batch_search(cmm_url = paste0("http://ceumass.eps.uspceu.es/mediator/api/v3/",
  "advancedbatch"), chemical_alphabet = "all", modifiers_type = "none",
  metabolites_type = "all-except-peptides", databases = "[\"hmdb\"]",
  masses_mode = "mz", ion_mode = "positive", adducts = "[\"all\"]",
  deuterium = "false", tolerance = "7.5", tolerance_mode = "ppm",
  masses = "[400.3432, 288.2174]", all_masses = "[]",
  retention_times = "[18.842525, 4.021555]",
  all_retention_times = "[]",
  composite_spectra = paste0("[[ { \"mz\": 400.3432, \"intensity\": 307034.88 }, ",
  "{ \"mz\": 311.20145, \"intensity\": 400.03336 } ] ]"))
```

Arguments

cmm_url	'http://ceumass.eps.uspceu.es/mediator/api/v3/advancedbatch' or your local one
chemical_alphabet	"CHNOPS", "CHNOPSCL", "ALL"
modifiers_type	"none", "NH3", "HCOO", "CH3COO", "HCOONH3", "CH3COONH3"
metabolites_type	"all-except-peptides", "only-lipids", "all-including-peptides"
databases	"all", "all-except-mine", "HMDB", "LipidMaps", "Metlin", "Kegg", "in-house", "mine"
masses_mode	"neutral", "mz"
ion_mode	"positive", "negative"
adducts	for positive mode ["M+H", "M+2H", "M+Na", "M+K", "M+NH4", "M+H-H2O"] for negative mode ["M-H", "M+Cl", "M+FA-H", "M-H-H2O"], for neutral ["M"]

```

deuterium      boolean 'true' 'false'
tolerance      double (Range: [0..100])
tolerance_mode "ppm", "mDa"
masses         double
all_masses     array of doubles
retention_times
               double
all_retention_times
               array of doubles
composite_spectra
               array of arrays of spectra_object

```

Value

If all inputs are all correctly formatted, a dataframe will be returned for the result.

Author(s)

Yaoxiang Li <yl1814@georgetown.edu>
 Georgetown University, USA
 License: GPL (>= 3)

Examples

```

df <- advanced_batch_search(
  cmm_url      = paste0(
    'http://ceumass.eps.uspceu.es/mediator/api/v3/',
    'advancedbatch'),
  chemical_alphabet = 'all',
  modifiers_type   = 'none',
  metabolites_type = 'all-except-peptides',
  databases        = ['hmdb'],
  masses_mode      = 'mz',
  ion_mode         = 'positive',
  adducts          = ['all'],
  deuterium        = 'false',
  tolerance        = '7.5',
  tolerance_mode   = 'ppm',
  masses           = '[400.3432, 288.2174]',
  all_masses       = '[]',
  retention_times  = '[18.842525, 4.021555]',
  all_retention_times = '[]',
  composite_spectra = paste0(
    '[[{ "mz": 400.3432, "intensity": 307034.88 }, ',
    '{ "mz": 311.20145, "intensity": 400.03336 }]]'
  ))

```

batch_search

*Encapsulation of CEU Mass Mediator batch search API***Description**

batch_search returns the dataframe of all the database search results. using the following code to install the dependencies: `install.packages(c("httr", "progress", "RJSONIO"))`

Usage

```
batch_search(cmm_url = "http://ceumass.eps.uspceu.es/mediator/api/v3/batch",
  metabolites_type = "all-except-peptides",
  databases = "[\`all-except-mine\`]", masses_mode = "mz",
  ion_mode = "positive", adducts = "[\`M+H\`,\`M+Na\`]",
  tolerance = 10, tolerance_mode = "ppm", unique_mz)
```

Arguments

cmm_url	'http://ceumass.eps.uspceu.es/mediator/api/v3/batch' or your local API Endpoint
metabolites_type	"all-except-peptides", "only-lipids", "all-including-peptides"
databases	"all", "all-except-mine", "HMDB", "LipidMaps", "Metlin", "Kegg", "in-house", "mine"
masses_mode	"neutral", "mz"
ion_mode	"positive", "negative"
adducts	for positive mode [M+H, M+2H, M+Na, M+K, M+NH4, M+H-H2O]
tolerance	double (Range: [0..100])
tolerance_mode	"ppm", "mDa"
unique_mz	An array of unique m/zs

Value

dataframe for search results

Author(s)

Yaoxiang Li <y1814@georgetown.edu>

Georgetown University, USA

License: GPL (>= 3)

Examples

```
## Not run:
batch_search(c(670.4623, 1125.2555, 602.6180))

## End(Not run)
```

```
create_advanced_batch_body
```

Create POST request Body for batch search

Description

create_advanced_batch_body returns a string of advanced search POST request body.

Usage

```
create_advanced_batch_body(chemical_alphabet = "all",
  modifiers_type = "none", metabolites_type = "all-except-peptides",
  databases = "[\\hmdb\\]", masses_mode = "mz",
  ion_mode = "positive", adducts = "[\\all\\]",
  deuterium = "false", tolerance = "7.5", tolerance_mode = "ppm",
  masses = "[400.3432, 288.2174]", all_masses = "[ ]",
  retention_times = "[18.842525, 4.021555]",
  all_retention_times = "[ ]",
  composite_spectra = paste0("[[ { \"mz\": 400.3432, \"intensity\": 307034.88 }, ",
    "{ \"mz\": 311.20145, \"intensity\": 400.03336 } ] ]"))
```

Arguments

chemical_alphabet	"CHNOPS", "CHNOPSCL", "ALL"
modifiers_type	"none", "NH3", "HCOO", "CH3COO", "HCOONH3", "CH3COONH3"
metabolites_type	"all-except-peptides", "only-lipids", "all-including-peptides"
databases	"all", "all-except-mine", "HMDB", "LipidMaps", "Metlin", "Kegg", "in-house", "mine"
masses_mode	"neutral", "mz"
ion_mode	"positive", "negative"
adducts	for positive mode ["M+H", "M+2H", "M+Na", "M+K", "M+NH4", "M+H-H2O"] for negative mode ["M-H", "M+Cl", "M+FA-H", "M-H-H2O"], for neutral ["M"]
deuterium	boolean 'true' 'false'
tolerance	double (Range: [0..100])

```

tolerance_mode "ppm", "mDa"
masses         double
all_masses    array of doubles
retention_times
              double
all_retention_times
              array of doubles
composite_spectra
              array of arrays of spectra_object

```

Value

If all inputs are all correctly formatted, a dataframe will be returned for the result.

Author(s)

Yaoxiang Li <y1814@georgetown.edu>
 Georgetown University, USA
 License: GPL (>= 3)

create_batch_body *Create POST request Body for batch search*

Description

create_batch_body returns a string of a POST request body.

Usage

```

create_batch_body(metabolites_type = "all-except-peptides",
  databases = "[\all-except-mine\]", masses_mode = "mz",
  ion_mode = "positive", adducts = "[\M+H\,\M+Na\]",
  tolerance = 10, tolerance_mode = "ppm", unique_mz)

```

Arguments

```

metabolites_type  "all-except-peptides", "only-lipids", "all-including-peptides"
databases         "all", "all-except-mine", "HMDB", "LipidMaps", "Metlin", "Kegg", "in-house",
                 "mine"
masses_mode       "neutral", "mz"
ion_mode          "positive", "negative"
adducts           for positive mode [M+H, M+2H, M+Na, M+K,M+NH4, M+H-H2O]
tolerance         double (Range: [0..100])
tolerance_mode    "ppm", "mDa"
unique_mz         An array of unique m/zs

```

Value

If all inputs are all correctly formatted, a string of a POST request will be returned for the result.

Author(s)

Yaoxiang Li <yl1814@georgetown.edu>

Georgetown University, USA

License: GPL (>= 3)

Examples

```
batch_body <- create_batch_body('all-except-peptides',  
                                ["all-except-mine"],  
                                'mz',  
                                'positive',  
                                ["M+H", "M+Na"],  
                                10,  
                                'ppm',  
                                c(670.4623, 1125.2555, 602.6180))
```

```
batch_body <- create_batch_body('all-except-peptides',  
                                ["all-except-mine"],  
                                'mz',  
                                'negative',  
                                ["M-H", "M+Cl"],  
                                10,  
                                'ppm',  
                                c(670.4623, 1125.2555, 602.6180))
```

```
## Not run:  
create_batch_body(c(670.4623, 1125.2555, 602.6180))
```

```
## End(Not run)
```

create_msms_body	<i>Create MS/MS search POST request body</i>
------------------	--

Description

create_msms_body returns a string of a POST request body.

Usage

```
create_msms_body(ion_mass, ms_ms_peaks, precursor_ion_tolerance = 500,  
                 precursor_ion_tolerance_mode = "mDa", precursor_mz_tolerance = 1000,  
                 precursor_mz_tolerance_mode = "mDa", ion_mode = "positive",  
                 ionization_voltage = "all", spectra_types = "experimental")
```

Arguments

ion_mass ion_mass
ms_ms_peaks ms_ms_peaks
precursor_ion_tolerance
 precursor_ion_tolerance
precursor_ion_tolerance_mode
 precursor_ion_tolerance_mode
precursor_mz_tolerance
 precursor_mz_tolerance
precursor_mz_tolerance_mode
 precursor_mz_tolerance_mode
ion_mode ion_mode
ionization_voltage
 ionization_voltage
spectra_types spectra_types

Value

If all inputs are all correctly formatted, a string of a POST request will be returned for the result.

Author(s)

Yaoxiang Li <y1814@georgetown.edu>
Georgetown University, USA
License: GPL (>= 3)

msms_search

Encapsulation of CEU Mass Mediator batch search API.

Description

batch_search returns the dataframe of all the database search results. using the following code to install the dependencies: `install.packages(c("httr", "progress", "RJSONIO"))`

Usage

```
msms_search(ion_mass, ms_ms_peaks, precursor_ion_tolerance = 100,  
          precursor_ion_tolerance_mode = "mDa", precursor_mz_tolerance = 500,  
          precursor_mz_tolerance_mode = "mDa", ion_mode,  
          ionization_voltage = "all", spectra_types = "experimental",  
          cmm_url = "http://ceumass.eps.uspceu.es/mediator/api/msmssearch")
```


Arguments

```
ion_mass      ion_mass
ms_ms_peaks   ms_ms_peaks
precursor_ion_tolerance
               precursor_ion_tolerance
precursor_ion_tolerance_mode
               precursor_ion_tolerance_mode
precursor_mz_tolerance
               precursor_mz_tolerance
precursor_mz_tolerance_mode
               precursor_mz_tolerance_mode
ion_mode      ion_mode
ionization_voltage
               ionization_voltage
spectra_types spectra_types
cmm_url       "http://ceumass.eps.uspceu.es/mediator/api/msmssearch" or your local one
```

Value

If all inputs are all correctly formatted, a dataframe will be returned for the result.

Author(s)

Yaoxiang Li <yl1814@georgetown.edu>
Georgetown University, USA
License: GPL (>= 3)

Examples

```
ms_ms_peaks = matrix(
c(40.948, 0.174,
  56.022, 0.424,
  84.37, 53.488,
  101.50, 8.285,
  102.401, 0.775,
  129.670, 100.000,
  146.966, 20.070),
ncol = 2,
byrow = TRUE)

msms_search(ion_mass = 147, ms_ms_peaks = ms_ms_peaks, ion_mode = 'positive')
```

Index

[advanced_batch_search](#), 2

[batch_search](#), 4

[create_advanced_batch_body](#), 5

[create_batch_body](#), 6

[create_msms_body](#), 7

[msms_search](#), 8