Genetic Collections Documentation

Release 0.1.6

Mike Trizna

Jul 07, 2021

Contents

Contents:

CHAPTER 1

Genetic Collections

A Python library for connecting genetic records with specimen data.

1.1 Installation

This software requires a working installation of Python 3.5 or higher. Your Python installation should come with a command-line tool called "pip", which is used to download packages from PyPI, the Python Package Index. Run the command below, and you should be good to go!

```
pip install genetic_collections
```

1.2 Command Line Usage

The installation from pip should also install several command line programs that act as wrappers for the code contained here.

Here are the available command line tools:

- ncbi_inst_search

```
$ ncbi_inst_search "Smithsonian"

6 matching results found.
 Fetching biocollection entries.
 [
   {
```

(continues on next page)

(continued from previous page)


```
    "Collection Type": "museum",
    "gb_count": 20697,
    "Country": "USA",
    "Institution Code": "USNM",
    "NCBI Link": "https://www.ncbi.nlm.nih.gov/biocollections/53",
    "Institution Name": "National Museum of Natural History, Smithsonian Institution"
  },
  {
    "Collection Type": "herbarium",
    "gb_count": 5269,
    "Country": "USA",
    "Institution Code": "US",
    "NCBI Link": "https://www.ncbi.nlm.nih.gov/biocollections/7399",
    "Institution Name": "Smithsonian Institution, Department of Botany"
  },
  ...
```


- gb_search

```
$ gb_search -inst_code USNM

Your search found 20697 hits in GenBank
You can see you search results online at
https://www.ncbi.nlm.nih.gov/nuccore/?term=%22collection+USNM%22%5Bprop%5D
```

- gb_fetch
- bold_inst_search
- bold_search
- bold_fetch

1.3 Python Library Usage

The best way to illustrate how the Python library can be used is to view the example workflow in the Jupyter notebook in the "examples" directory.

1.4 How to contribute

Imposter syndrome disclaimer: I want your help. No really, I do.

There might be a little voice inside that tells you you're not ready; that you need to do one more tutorial, or learn another framework, or write a few more blog posts before you can help me with this project.

I assure you, that's not the case.

This project has some clear Contribution Guidelines and expectations that you can read here (link).

The contribution guidelines outline the process that you'll need to follow to get a patch merged. By making expectations and process explicit, I hope it will make it easier for you to contribute.

And you don't just have to write code. You can help out by writing documentation, tests, or even by giving feedback about this work. (And yes, that includes giving feedback about the contribution guidelines.)

Thank you for contributing!

1.5 Next Steps

- Incorporate MIXS standards
- Add the ability to translate GenBank and BOLD results to DwC format in order to compare
- Add iDigBio and GBIF APIs as data sources for specimen data (and GenBank accessions)

1.6 Credits

"How to contribute" was taken from https://github.com/adriennefriend/imposter-syndrome-disclaimer.

This package was created with Cookiecutter and the audreyr/cookiecutter-pypackage project template.

CHAPTER 2

API Documentation

2.1 NCBI

genetic_collections.ncbi_functions.**gb_fetch_from_id_list**(*id_list*, *batch_size=250*, *api_key=None*)

Orchestrates making calls to the NCBI efetch service, and passes off the XML to the parse_fetch_results function.

Parameters

- **id_list** (*list*) – List of GenBank IDs – this can be either GIs or Accessions
- **batch_size** (*int*) – The number of results to request at a time – the higher the better, but too large of a result set causes errors
- **api_key** (*str*) – A personal NCBI API key, obtained by creating an NCBI account. An API key is not required, but supplying one will increase API call rate to NCBI from three per second to 10 per second.

Return type list of dictionaries

genetic_collections.ncbi_functions.**ncbi_taxonomy**(*gb_fetch_results*, *batch_size=250*, *api_key=None*)

Orchestrates making calls to the NCBI efetch service, querying the NCBI Taxonomy database for a list of NCBI taxids. Passes off the XML to the ncbi_parse_taxonomy_xml function.

Parameters

- **gb_fetch_results_list** – The unprocessed results of a previous gb_fetch_from_id_list
- **batch_size** (*int*) – The number of results to request at a time – the higher the better, but too large of a result set causes errors
- **api_key** (*str*) – A personal NCBI API key, obtained by creating an NCBI account. An API key is not required, but supplying one will increase API call rate to NCBI from three per second to 10 per second.

Return type list of dictionaries

- search

Python Module Index

g

Index

G

N