`biacpype` User Manual

Adcock Lab
Center for Cognitive Neuroscience, Duke University

September 5, 2018

Contents

1 Overview

biacpype serves as a pipeline for converting raw fMRI data from Brain Imaging & Analysis Center (BIAC) at Duke University to the new standard Brain Imaging Data Structure (BIDS) format, and supporting furthur analysis. The main parts of biacpype contains:

- util: BIAC dataset validation, logging
- biac2bids: conversion from BIAC to BIDS
- ...

The structure of this repository is:

```
|-biacpype // serves as the library
|    |
|    |_biac2bids
|    |_ ...
|
|-scripts  // scripts for running a variety of pipelines
|    |
|    |_convert_to_bids.py
|    |_ ...
|-tutorial // a tutorial notebook
|
|-docs     // documentation (under construction)
    |_ ...
```

To install dependencies of this repo, run

```
pip install -r requirements.txt
```

The following section will introduce main modules of biacpype, and the corresponding scripts for using the pipelines. All scripts are described in **Scripts** subsections.

If you have any questions, please email Preston Jiang at linxing.jiang@duke.edu or prestonj@cs.washington.edu

2 util

This is the module which serves as the utility module. It contains validation functionality to let you validate your dataset from BIAC to check if it is ready to be converted.

2.1 Dataset Structure

The pipeline makes the following assumptions on the format of your raw data from BIAC:

```
|-Data
|    |
|    |-Func
|    |  |
|    |  |-<[date_]subject>
|    |  |        |
|    |  |        |-<biac5_subject_task_run>.bxh
|    |  |        |-<biac5_subject_task_run>.nii.gz
|    |  |        |-...
|    |  |        |-series_order_note.tsv
|    |  |-...
|    |
|    |-Anat
|       |-<[date_]subject>
|       |        |
|       |        |-<biac5_subject_task_run>.bxh
|       |        |-<biac5_subject_task_run>.nii.gz
|       |        |-...
|       |        |-series_order_note.tsv
|       |-...
|
|-biac_id_mapping.tsv
```

Explanations:

- `Data` folder has to contain `Func` and `Anat`, and they must have the exact same folders
- Subfolders in `Func` and `Anat` are in format `[date_]subject` where `[date_]` is optional. E.g. `19354` and `20140101_19354` are both acceptable.
- Each file in `Func` and `Anat` must in format `biac5_subject_task_run`. Usually, `task` is a single digit number, `run` is two-digit. E.g. `biac5_19354_4_01.bxh`
- Each subfolder in `Func` and `Anat` **must contain** a `series_order_note.tsv` to tell the pipeline what each `task` number stands for. E.g. 4 stands for "TRAIN". Requirements for this file are later explained.
- In the same folder containing `Data`, there **must contain** a `biac_id_mapping.tsv` which tells the pipeline the mapping from BIAC_ID (e.g. 19354) to the session name (e.g. Session-1) and the Real_ID used by your lab (e.g. 101). Requirements for this file are later explained.

Requirements on `series_order_note.tsv` are as follows:

4	LOCALIZER
7	TRAIN1
...	...

Note:

- The values must be **tab** separated.
- The first column serves as the primary key (they must be unique)
- Task numbers in the first column ***must*** match that in the imaging files. For example, a "4" in `translation_file` and a "004" in the imaging file will result in an error.
- The first column must be task code, and the second column must be the task name. There can only be two columns
- If you have fmap data task code, be sure to name the translation **"fmap"** exactly. In this way, the data will be put in `fmap` folder after BIDS conversion, not `anat` folder.

Requirements on `biac_id_mapping.tsv` are as follows:

BIAC_ID	[Session]	Real_ID
19354	SRM	101
19338	SPM	102
19368	SPM	101
...	...	...

Note:

- The values must be **tab** separated.
- The first column serves as the primary key (they must be unique)
- The headers must follow the rules (watch letter cases)!
- If your experiment does not have multiple sessions, you can ignore the `Session` column

2.2 Validation

The requirements the module validates are:

1. Basic structure: check folder has `Data`, `biac_id_mapping.tsv`, Data has `Anat`, `Func`, both `Anat` and `Func` contain the same folders, and each of these folders contains `series_order_note.tsv`

2. `biac_id_mapping_file`: check `biac_id_mapping.tsv` is valid
3. `all_series_order_note_files`: check all of the `series_order_note.tsv` files in each subfolder is valid
4. Data matching: check each `.bxh` file is matched with `.nii.gz` file, and has valid task code translation which can be found in `series_order_note.tsv`

2.3 Scripts

The script associated with this module is `scripts/validate_biac_study_folder.py`. The only thing you need to change is the `STUDY_PATH`.

If there are errors, they will be printed out to console after validation finishes. Otherwise, you will see the following printed out:

```
Your study path passed validation!  You are now ready for conversion
```

You can also check out all the logs produced under `biacpype/logs/validation.log`

3 `biac2bids`

`biac2bids` module is the pipeline for converting raw data from BIAC in forms of `bxh` and `nifti` to BIDS format. The workflow is as follows:

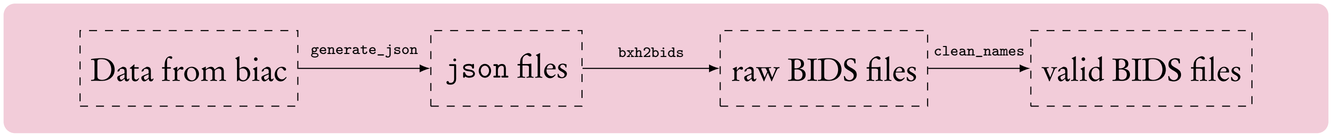

Figure 1: `biac2bids` workflow

3.1 Scripts

The script associated with this module is `scripts/convert_to_bids.py`. There are four parameters the user has to toggle. They are:

- `STUDY_PATH`: the path to your study file (which contains `Data` and `bids_id_mapping.csv`)
- `JSON_OUTPUT_PATH`: the path where the user wants the json files to be saved
- `BIDS_PATH`: the path where the user wants the new BIDS format data to be saved
- `LOG_PATH`: the path where the user wants the logs to be saved

3.2 Common Errors

1. `Error:root:scan description not found in template file!`

   This is because the scan description (printed out next line) is not saved in template file. To fix this, add the description to `biacpype/biac2bids/bxh2bids/info_field_files/psd_types.json` following its format.

4 `helper`

For now, the `helper` contains one helper function which creates `series_order_note.tsv` automatically from user script input.