

Bidsme

An unuversal dataset bidsifier

Nikita BELIY¹

¹Liege University, Belgium

04 March, 2020



- 1 Introduction
- 2 Basic usage
- 3 bidsmapi configuration file
- 4 Plugins

Introduction

Bidsme

[CRC](#) [GitLab](#), [GitHub](#)

- Object-oriented version of *[Bidscoin](#)*
- Not just renames but fully bidsifies
- Follows BIDS [v1.2.0](#)
- Allow deviations from BIDS
- Supports plugins
- Not limited to predefined modalities/data formats

Example dataset

[CRC](#) [GitLab](#), [GitHub](#)

- Inspired by real-life dataset
- Contains needed scripts and configurations

Command line interface

- `python3 bidsme.py` *<command>* *<source>* *<destination>* *<options>*
 - `bidsme.py` – main script
 - *<command>* – select different process steps
 - *<source>* – path to source dataset
 - *<destination>* – path to destination dataset
 - *<options>* – additional options

Online help

- `python3 bidsme.py --help`
- `python3 bidsme.py` *<command>* `--help`

General workflow

Datasets

- **Source dataset**

- Original, unbidsified dataset
- Only readed, not modified by *bidsme*

- **Prepared/Renamed dataset**

- Temporary dataset for user intervention
- Error corrections, data completion, anonymisations etc.
- Can be tested, erased, recreated

- **Bidsified dataset**

- Final, bidsified dataset

General workflow

Processing steps/commands

- ① **map** (once):
 - creates a bidsification schema
- ① **prepare**:
 - Put source dataset into standardized folder structure
 - Identifies subjects, sessions, modalities and series
 - (Plugin) Retrieve metadata from exterior sources
- ② **process** (optional):
 - Test for various errors and inconsistencies
 - (Plugin) Transform prepared dataset (e.g. 3D to 4D merging)
- ③ **bidsify**:
 - Bidsifies prepared dataset
 - Manages *participants.tsv* and *scans.tsv*
 - (Plugin) Creates behavioural and derived data

Basic usage

Source dataset

bidsme_examples/example1/source

- unmodified dataset
- intuitive structure
[subject/][session/][format/]data.nii
- subject, session and format folders optional

The diagram shows a directory tree for a BIDS source dataset. The root is 'source/'. It contains a subdirectory '001/'. Inside '001/' is 's01512/'. Inside 's01512/' is 'inp/'. Inside 'inp/' are 'FCsepNBack.tsv' and 'VAS.tsv'. Also inside 's01512/' is a subdirectory 'nii/'. Inside 'nii/' are various files with names like 'f1512-0002-00001-000001-01.json', 'f1512-0002-00001-000001-01.nii', etc. Annotations with red arrows point to '001/' (Subject), 's01512/' (Session), and the 'nii/' directory (MRI images).

```
source/
|-- 001/
|   |-- s01512/
|       |-- inp/
|           |-- FCsepNBack.tsv
|           |-- VAS.tsv
|       |-- nii/
|           |-- f1512-0002-00001-000001-01.json
|           |-- f1512-0002-00001-000001-01.nii
|           |-- f1512-0002-00002-000002-01.json
|           |-- f1512-0002-00002-000002-01.nii
|           |-- f1512-0002-00003-000003-01.json
|           |-- f1512-0002-00003-000003-01.nii
|           |-- f1512-0002-00004-000004-01.json
|           |-- f1512-0002-00004-000004-01.nii
|           |-- f1512-0002-00005-000005-01.json
|           |-- f1512-0002-00005-000005-01.nii
|           |-- f1512-0004-00001-000001-01.json
|           |-- f1512-0004-00001-000001-01.nii
|           |-- f1512-0004-00002-000002-01.json
|           |-- f1512-0004-00002-000002-01.nii
|           |-- f1512-0004-00003-000003-01.json
|           |-- f1512-0004-00003-000003-01.nii
|           |-- f1512-0004-00004-000004-01.json
|           |-- f1512-0004-00004-000004-01.nii
|           |-- f1512-0004-00005-000005-01.json
|           |-- f1512-0004-00005-000005-01.nii
```

Basic usage

Prepared dataset

bidsme_examples/example1/renamed

- Created by **prepare** command
- bids-like folder structure
sub-subject/ses-session/format/series/data.nii
- subject, session determined from source dataset
can be modified by plugin

The diagram shows the directory structure of the 'renamed/' folder. Red boxes and arrows highlight specific parts:

- A box around 'log/' and 'coinsort.err' points to the text 'Log files for preparation'.
- A box around 'participants.json' and 'participants.tsv' points to 'Subjects info'.
- A box around 'sub-001/' and 'ses-s01512/' points to 'Subject, session and data type'.
- Arrows point to '001-localizer/' and '002-cmrr mbep2d_bold_mb2_invertpe/' with the label 'Series No and Id'.

```

renamed/
|-- .gitignore
|-- log/
|   |-- coinsort.err
|   |-- coinsort.log
|-- participants.json
|-- participants.tsv
|-- sub-001/
|   |-- ses-s01512/
|   |   |-- MRI/
|   |   |-- 001-localizer/
|   |       |-- s1512-0001-00001-000001-01.json
|   |       |-- s1512-0001-00001-000001-01.nii
|   |       |-- s1512-0001-00001-000002-01.json
|   |       |-- s1512-0001-00001-000002-01.nii
|   |       |-- s1512-0001-00001-000003-01.json
|   |       |-- s1512-0001-00001-000003-01.nii
|   |       |-- 002-cmrr mbep2d_bold_mb2_invertpe/
|   |           |-- f1512-0002-00001-000001-01.json
|   |           |-- f1512-0002-00001-000001-01.nii
|   |           |-- f1512-0002-00002-000002-01.json
|   |           |-- f1512-0002-00002-000002-01.nii
|   |           |-- f1512-0002-00003-000003-01.json
    
```

Example:

```
python3 bidsme.py prepare source renamed -r nii=MRI
```


Basic usage

Bidsified dataset

bidsme_examples/example1/bids

- Created by **bidsify**
- Fully bidsified dataset:
 - subject list (participants.tsv)
 - session if defined (ses-s01512)
 - scan list (scans.tsv)
 - modalities and filenames adapted
 - each scan has bids json with metadata
- additional data created by plugins

```
bids/
-- .gitignore
-- code/
  -- bidscoin/
    -- bidsmap.yaml.swp
    -- bidsmap.yaml
    -- log/
      -- bidscoiner.err
      -- bidscoiner.log
      -- bidsmapper.err
      -- bidsmapper.log
  -- participants.json
  -- participants.tsv
  -- sub-001/
    -- ses-s01512/
      -- anat/
        -- sub-001_ses-s01512_FLAIR.json
        -- sub-001_ses-s01512_FLAIR.nii
        -- sub-001_ses-s01512_T1w.json
        -- sub-001_ses-s01512_T1w.nii
      -- fmap/
        -- sub-001_ses-s01512_magnitude1.json
        -- sub-001_ses-s01512_magnitude1.nii
```

Annotations:

- bidsmap file**: points to `bidsmap.yaml`
- log files for bidsification**: points to the `log/` directory
- bidsified data**: points to the `sub-001/ses-s01512/` directory

Example:

```
python3 bidsme.py bidsify renamed bids
-b resources/map/bidsmap_noPlugin.yaml
```

bidsmap configuration file

- yaml configuration file
- Tells to **bidsme** how treat dataset
 - how identify recording
 - how bidsify recording
- Created from template by map
 - template in *bidsme/heuristics*
 - one per institute
 - should cover all protocols

bidsmap configuration file

Body

resources/map/bidsmap.yaml

Nested dictionary:

- Type: type of recording (MRI, EEG, PET etc. . .)
MRI and EEG implemented
 - File format: nifti, dicom, etc. . .
dicom, nifti (hMRI) and BrainVision implemented
 - modality: anat, func, eeg etc. . .
corresponds to BIDS modality folder

```
MRI:
  nifti SPM12:
    ignore :
      - provenance: /home/beliy/Works/bidscoin/test/renamed/sub-003/s
        checked: true
        suffix: ''
        attributes:
          ProtocolName: 'localizer'
        bids: !!omap []
        json: !!omap []
```

bidsmap configuration file

Runs

<resources/map/bidsmap.yaml>

Runs: elements to check individual recordings

- **provenance**: recording from which this entry is generated
- **example**: bidsified name from provenance file
- **template**: true if entry retrieved from template
- **checked**: true if user checked this entry
- **attributes**: dictionary of metadata to check files
- **suffix** to apply
- **bids**: dictionary of bids entities to generate
- **json**: list of additional entries for json sidecar

```
- provenance: /home/beliy/Works/bidscoin/test/renamed/sub-003/ses-STR00P/MRI/  
  example: fmap/sub-003_ses-STR00P_acq-HeadPDw_B1minus  
  checked: true  
  suffix: B1minus  
  attributes:  
    ProtocolName: 'a1_mtf1ash3d_sensArray'  
    SeriesDescription: 'PDw'  
  bids: !!omap  
    - acq: HeadPDw  
    - mod: -  
  json: !!omap  
    - EchoTime: <time3:EchoTime>  
    - IntendedFor:  
      - "anat/<<subject>>_<<session>>_acq-<<bids.mod>>_echo-* MPM.nii"
```

bidsmap configuration file

Runs

<resources/map/bidsmap.yaml>

Runs: elements to check individual recordings

- provenance: recording from which this entry is generated
- example: bidsified name from provenance file
- **template**: true if entry retrieved from template
- **checked**: true if user checked this entry
- attributes: dictionary of metadata to check files
- suffix to apply
- bids: dictionary of bids entities to generate
- json: list of additional entries for json sidecar

```
- provenance: /home/beliy/Works/bidscoin/test/renamed/sub-003/ses-STR00P/MRI/
  example: fmap/sub-003_ses-STR00P_acq-HeadPDw_B1minus
  checked: true
  suffix: B1minus
  attributes:
    ProtocolName: 'a1_mtf1ash3d_sensArray'
    SeriesDescription: 'PDw'
  bids: !!omap
    - acq: HeadPDw
    - mod: ~
  json: !!omap
    - EchoTime: <time3:EchoTime>
    - IntendedFor:
      - "anat/<<subject>>_<<session>>_acq-<<bids.mod>>_echo-* MPM.nii"
```

bidsmap configuration file

Runs

[resources/map/bidsmap.yaml](#)

Runs: elements to check individual recordings

- provenance: recording from which this entry is generated
- example: bidsified name from provenance file
- template: true if entry retrieved from template
- checked: true if user checked this entry
- **attributes**: dictionary of metadata to check files
- suffix to apply
- bids: dictionary of bids entities to generate
- json: list of additional entries for json sidecar

```
- provenance: /home/beliy/Works/bidscoin/test/renamed/sub-003/ses-STR00P/MRI/
  example: fmap/sub-003_ses-STR00P_acq-HeadPDw_B1minus
  checked: true
  suffix: B1minus
  attributes:
    ProtocolName: 'a1_mtflash3d_sensArray'
    SeriesDescription: 'PDw'
  bids: !!omap
    - acq: HeadPDw
    - mod: ~
  json: !!omap
    - EchoTime: <time3:EchoTime>
    - IntendedFor:
      - "anat/<<subject>>_<<session>>_acq-<<bids.mod>>_echo-* MPM.nii"
```

bidsmap configuration file

Runs

<resources/map/bidsmap.yaml>

Runs: elements to check individual recordings

- provenance: recording from which this entry is generated
- example: bidsified name from provenance file
- template: true if entry retrieved from template
- checked: true if user checked this entry
- attributes: dictionary of metadata to check files
- **suffix**: modality label
- **bids**: dictionary of bids entities to generate
- **json**: list of additional entries for json sidecar

```
- provenance: /home/beliy/Works/bidscoin/test/renamed/sub-003/ses-STR00P/MRI/
  example: fmap/sub-003_ses-STR00P_acq-HeadPDw_B1minus
  checked: true
  suffix: B1minus
  attributes:
    ProtocolName: 'a1_mtflash3d_sensArray'
    SeriesDescription: 'PDw'
  bids: !!omap
    - acq: HeadPDw
    - mod: ~
  json: !!omap
    - EchoTime: <time3:EchoTime>
    - IntendedFor:
      - "anat/<<subject>>_<<session>>_acq-<<bids:mod>>_echo-*_MPM.nii"
```

Dynamic field retrieval

field	transforms to	example
text	itself	abc_ → abc_
⟨text⟩	value in header	⟨EchoTime⟩ → 2.2
⟨func:text⟩	func ^(a) (value in header)	⟨time3:EchoTime⟩ → 0.0022
⟨⟨text⟩⟩	recording parameter ^(a)	⟨⟨subject⟩⟩ → sub-003
⟨⟨bids:text⟩⟩	bids parameter	⟨⟨bids:acq⟩⟩ → HeadPDw
⟨⟨sub_tsv:text⟩⟩	subject parameter	⟨⟨sub_tsv:age⟩⟩ → 22
⟨⟨rec_tsv:text⟩⟩	recording parameter	⟨⟨rec_tsv:acq_time⟩⟩ → 2018-09-18T15:10:15
⟨⟨placeholder⟩⟩	produce warning	

^aSee backup slides

Example:

```
anat/⟨⟨subject⟩⟩_⟨⟨session⟩⟩_acq-⟨⟨bids:acq⟩⟩_echo-*_MPM.nii
anat/sub-003_ses-STROOP_acq-HeadPDw_echo-*_MPM.nii
```


Bidsmap generation

map

- runs on prepared dataset
- creates or updates bidsmap file from template
- print errors/warnings for corrections

Workflow:

- 1 Remove 'template' status
- 2 Correct errors^(a)
- 3 Correct warnings^(a)
- 4 Set 'checked' status

^(a)See backup slides for errors/warnings

Plugins

- python3 file with pre-defined functions (entry points)
- each function called in specific place of code
- have full control on handled data
flexible but dangerous
- exceptions either by 'raise' or return code > 0
- skip current entity by return code < 0
- template/reference: bidscoin/plugins/template.py
- example: examples1/resources/plugins

Example preparation

[resources/plugins/rename_plugin.py](#)

Executed during data preparation

- InitEP:
loads excel table with list of subjects
- SubjectEP:
retrieve demographic info from given subject
- SessionEP:
retrieve session name from subject table
- SessionEndEP:
checks the validity of series
copy task logfile to output directory

Example preparation

Prepared data

resources/plugins/rename_plugin.py

Without plugin

*prepare source renamed -r
nii=MRI*

```
renamed/
|-- .gitignore
|-- log/
|   |-- coinsort.err
|   |-- coinsort.log
|-- participants.json
|-- participants.tsv
|-- sub-001/
|   |-- ses-s01512/
|   |-- MRI/
|       |-- 001-localizer/
|           |-- s1512-0001-00001-000001-01.json
|           |-- s1512-0001-00001-000001-01.nii
|           |-- s1512-0001-00001-000002-01.json
|           |-- s1512-0001-00001-000002-01.nii
|           |-- s1512-0001-00001-000003-01.json
|           |-- s1512-0001-00001-000003-01.nii
|       |-- 002-cmrr_mbep2d_bold_mb2_invertpe/
|           |-- f1512-0002-00001-000001-01.json
|           |-- f1512-0002-00001-000001-01.nii
|           |-- f1512-0002-00002-000002-01.json
|           |-- f1512-0002-00002-000002-01.nii
|           |-- f1512-0002-00003-000003-01.json
```

Log files for preparation →

Subjects info →

Subject, session and data type →

Series No and Id →

With plugin

*prepare source renamed -r
nii=MRI -p rename_plugin.py*

```
renamed/
|-- .gitignore
|-- log/
|   |-- coinsort.err
|   |-- coinsort.log
|-- participants.json
|-- participants.tsv
|-- sub-001/
|   |-- ses-HCL/
|   |-- MRI/
|       |-- 001-localizer/
|           |-- s1513-0001-00001-000001-01.json
|           |-- s1513-0001-00001-000001-01.nii
|           |-- s1513-0001-00001-000002-01.json
|           |-- s1513-0001-00001-000002-01.nii
|           |-- s1513-0001-00001-000003-01.json
|           |-- s1513-0001-00001-000003-01.nii
|       |-- 002-cmrr_mbep2d_bold_mb2_invertpe/
|           |-- f1513-0002-00001-000001-01.json
|           |-- f1513-0002-00001-000001-01.nii
|           |-- f1513-0002-00002-000002-01.json
|           |-- f1513-0002-00002-000002-01.nii
|           |-- f1513-0002-00003-000003-01.json
```

Exported subject data →

Retrieved session name →

Example preparation

Subject info

[resources/plugins/rename_plugin.py](#)

Without participants template

bidsify renamed bids

```
participant_id
sub-001
sub-002
sub-003
sub-004
renamed/participants.tsv (END)
```

With template

--part-template participants.json

participant_id	age	sex	education	group	handiness	paired
sub-001	n/a	n/a	n/a	n/a	n/a	n/a
sub-002	n/a	n/a	n/a	n/a	n/a	n/a
sub-003	n/a	n/a	n/a	n/a	n/a	n/a
sub-004	n/a	n/a	n/a	n/a	n/a	n/a

Pre-created columns, to fill manually

With plugin and template

-p bidsify_plugin.py --part-template participants.json

participant_id	age	sex	education	group	handiness	paired
sub-001	38	M	12	patient	n/a	sub-003
sub-002	32	F	10	patient	n/a	sub-004
sub-003	38	M	12	control	n/a	sub-001
sub-004	32	F	10	control	n/a	sub-002

Filled values by plugin

Example bidsification

resources/plugins/bidsify_plugin.py

Executed during bidsification

- SessionEP:
checks the validity of series and copy task log files
- SequenceEP:
retrieve the correct task for LCL/HCL sessions
- RecordingEP: sets 'SeriesDescription' accordingly to task

Example bidsification

Additional modality

resources/plugins/bidsify_plugin.py

Without plugin

bidsify renamed bids

```
bids/
|-- .gitignore
|-- code/
|   |-- bidscoin/
|   |   |-- bidsmap.yaml
|   |   |-- bidsmap_noPlugin.yaml
|   |   |-- log/
|   |       |-- bidscoiner.err
|   |       |-- bidscoiner.log
|   |       |-- bidsmapper.err
|   |       |-- bidsmapper.log
|-- participants.json
|-- participants.tsv
|-- sub-001/
|   |-- ses-s01512/
|   |   |-- anat/
|   |       |-- sub-001_ses-s01512_FLAIR.json
|   |       |-- sub-001_ses-s01512_FLAIR.nii
|   |       |-- sub-001_ses-s01512_T1w.json
|   |       |-- sub-001_ses-s01512_T1w.nii
|   |   |-- fmap/
|   |       |-- sub-001_ses-s01512_magnitude1.json
|   |       |-- sub-001_ses-s01512_magnitude1.nii
```

With plugin

bidsify renamed bids

-p bidsify_plugin.py

```
bids/
|-- .gitignore
|-- code/
|   |-- bidscoin/
|   |   |-- .bidsmap.yaml.swp
|   |   |-- bidsmap.yaml
|   |   |-- log/
|   |       |-- bidscoiner.err
|   |       |-- bidscoiner.log
|   |       |-- bidsmapper.err
|   |       |-- bidsmapper.log
|-- participants.json
|-- participants.tsv
|-- sub-001/
|   |-- ses-HCL/
|   |   |-- beh/
|   |       |-- sub-001_ses-HCL_task-rest_beh.json
|   |       |-- sub-001_ses-HCL_task-rest_beh.tsv
|   |       |-- sub-001_ses-HCL_task-rest_events.json
|   |       |-- sub-001_ses-HCL_task-rest_events.tsv
|   |   |-- dwi/
|   |       |-- sub-001_ses-HCL_acq-RF_dir-AP_run-1_dwi/
```

task log files created by plugin

Conclusion I

What works

- developed for Linux
- confirmed to work on Windows (PowerShell)
- NIFTI (hMRI) fully tested
- DICOM, generic NIFTI implemented and tested
- BrainVision implemented and tested

What to test

- Datasets outside CRC
- EEG/BrainVision

Conclusion II

Plans

- new formats (EDF, others?)
- plugin access to sidecar json
- BIDS version changing
- bidsified dataset versioning

dynamic values

Recording parameters

Recording parameters

field	value	notes and examples
<<subject>>	Subject Id	sub-001
<<session>>	Session Id	ses-LCL
<<serieNumber>>	No of Series	10
<<serie>>	Id of Series	gre_field_mapping
<<index>>	File index in series	sorted in alphabetic order
<<nfiles>>	N files in series	2
<<filename>>	current file name	s1512-0010-00001-000001-01.nii
<<suffix>>	bids modality suffix	magnitude1
<<modality>>	bids modality	fmap
<<module>>	File type	MRI
<<placeholder>>	itself	always produce warning

dynamic values

Attribute functions

Attribute functions

field	function	notes and examples
$\langle \text{timeN:field} \rangle$	$\times 10^{-N}$	$\langle \text{time3:EchoTime} \rangle = 0.0022$

bidsmapper warnings

Warning	correction
Placeholder found	Replace placeholder by corresponding value
Can't find attribute	check and correct attribute
Naming schema not BIDS	check BIDS schema and correct if needed
Looking into template	setting status 'checked' to true remove warning
Multiple runs	ignore at first, check attributes if raises again
	Tighten attributes to match only one run

Error	correction
Run not found	Retrieve recording in 'unknown.yaml'
	and create/correct corresponding run
YAML parsing error	correct bidsmap at given location