

OPEN
ARTICLE

The R package for DICOM to brain imaging data structure conversion

Niklas Wulms¹✉, Sven Eppe², Mahboobeh Dehghan-Nayyeri^{3,4}, Adam J. Streeter¹, Nadine Bonberg¹, Klaus Berger¹, Benedikt Sundermann^{1,4,5,6,7} & Heike Minnerup^{1,7}

The BIDSconvertR package is the first R-based tool for organizing magnetic resonance imaging (MRI) research data in accordance with the Brain Imaging Data Structure (BIDS) specification. Key features are the DICOM (Digital Imaging and Communications in Medicine) to NIfTI (Neuroimaging Informatics Technology Initiative) and NIfTI to BIDS conversion, the implementation of the BIDS Validator and a MRI data viewer to efficiently manage MRI neuroimaging data sets. The BIDSconvertR offers an interactive user dialogue and a graphical user interface. BIDS validation is facilitated by color-coding of the BIDS sequence-IDs. Data cleaning is simplified by the option of using regular expressions. The BIDSconvertR contributes to the growing efforts to improve reproducibility in neuroimaging research by facilitating researchers to share and organize data in a standardized and transparent manner.

Introduction

When it comes to effective research data management, there are various challenges to meet, including different data and folder structures, as well as naming conventions. The need for distinct specifications to prevent untidy data becomes increasingly important as data sets grow larger and more complex. Neuroimaging data and meta-data, for instance, are named and organized in a variety of ways across researchers and laboratories. As a result, data analysis pipelines (preprocessing or statistical) that have been developed for a specific data set can not simply be applied to other data sets, which increases efforts and impairs reproducibility. The DICOM format was a significant milestone in the standardization of medical neuroimage storage and communication. However, it is not human-readable and typically consists of many files per image. As a more workable option for scientific use the NIfTI standard was created. It can be directly converted from MRI DICOM data, has a human-readable format and consists of one file per image¹. More recently, the Brain Imaging Data Structure (BIDS) was created by the BIDS-consortium as a standardized method of storing and reporting neuroimaging data (MRI, MEG, and EEG) alongside corresponding behavioral or clinical data². The purpose of this specification is to facilitate research collaborations, methodological developments (such as pre-processing and analysis pipelines) and replication studies in the field of neuroimaging³. There are multiple tools (<https://bids.neuroimaging.io/benefits.html>) that can convert DICOM to NIfTI (such as `dcm2niix`⁴ that is used by our tool), NIfTI to BIDS, or both. However, advanced programming knowledge in Python or Matlab is often required to use the BIDS conversion software. Only recently, the first few tools have been released that present a user-friendly graphical user interface⁵.

Here, we introduce the ‘BIDSconvertR’⁶, the first package for BIDS conversion developed in R. It covers every step from DICOM input to BIDS output. The tool is suitable for users with less programming experience as it mainly works with pop-up boxes and graphical user interfaces. It converts DICOM to NIfTI with the widely distributed and validated `dcm2niix`⁴. The usability is optimized by an interactive user dialogue to select the input and output folder, the option to define the subject-ID or remove patterns with regular expressions, the color-coded validation of BIDS sequence-IDs and validation of the resulting data set with the BIDS-Validator⁷. It also features ‘Shiny BIDS’, an interactive MRI viewer based on papaya that allows users to effortlessly browse through the layered BIDS folder structure and to manually assess image quality.

¹Institute of Epidemiology and Social Medicine, University of Muenster, Muenster, Germany. ²State Cancer Registry of North Rhine-Westphalia, Bochum, Germany. ³Department of Psychosomatic Medicine and Psychotherapy, LVR Clinic, Medical Faculty of the Heinrich-Heine-University Duesseldorf, Duesseldorf, Germany. ⁴Clinic of Radiology, University Hospital Muenster, Muenster, Germany. ⁵Institute of Radiology and Neuroradiology, Evangelisches Krankenhaus, Medical Campus, University of Oldenburg, Oldenburg, Germany. ⁶Research Center Neurosensory Science, University of Oldenburg, Oldenburg, Germany. ⁷These authors contributed equally: Benedikt Sundermann, Heike Minnerup. ✉e-mail: wulms@uni-muenster.de

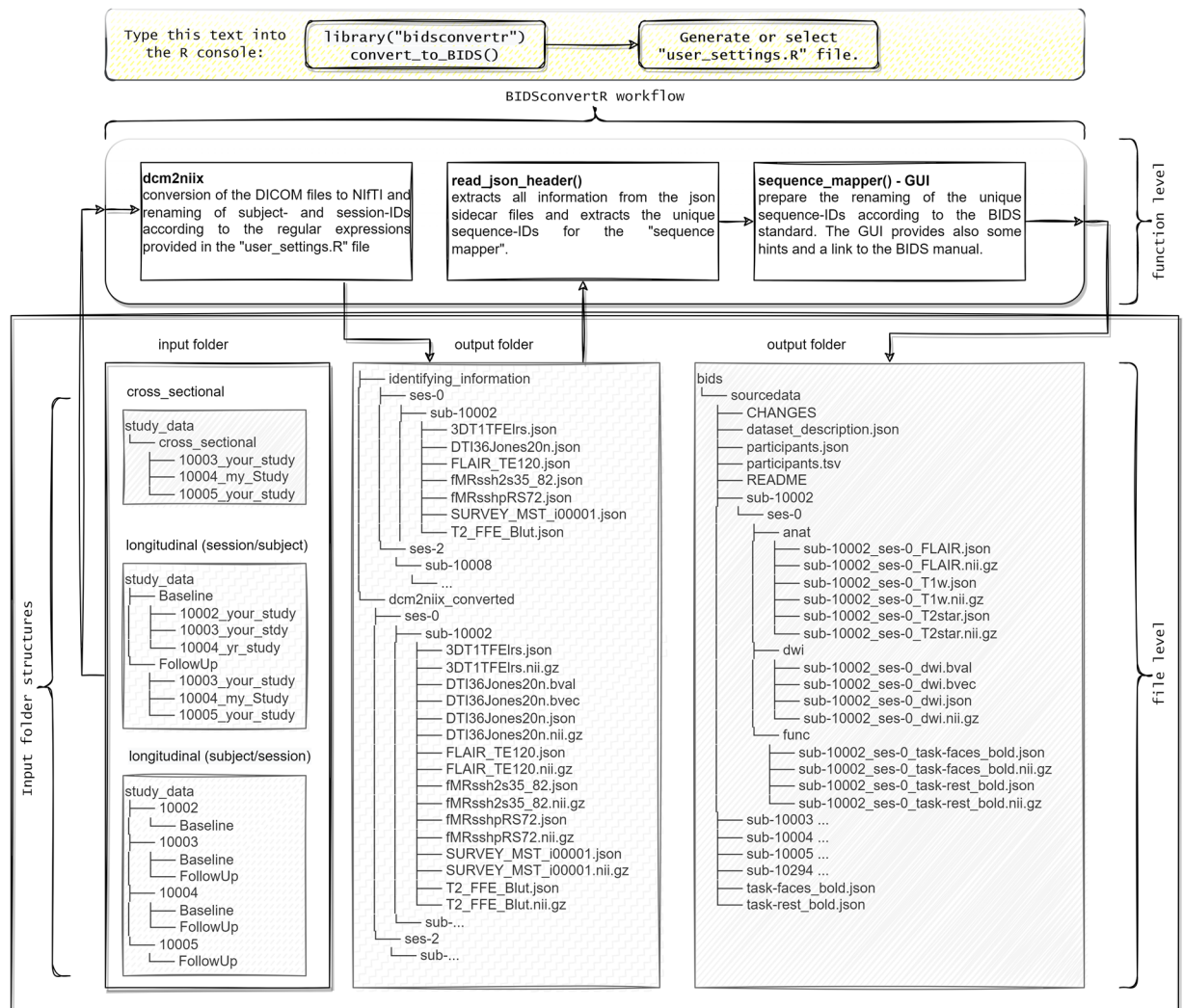


Fig. 1 Visualization of an example data flow and the ‘convert_to_BIDS()’ function. The input DICOM data will be indexed by subject-ID and session-ID. If the provided regular expressions match the data, all DICOM files will be converted and renamed according to BIDS to NIfTI. In the next step, all unique sequence-ID’s will be extracted and provided in an additional file. Now, the user has to use the “sequence mapper” graphical user interface to change these sequence-ID’s to BIDS output sequence names and types and choose whether or not they should be stored to BIDS.

This publication aims to introduce the tool to the scientific community by detailing the prerequisites, usage, functionality, and features of the ‘BIDScnvertR’⁶ (<https://github.com/bidsconvertr/bidsconvertr>). It is accompanied by a detailed documentation website (<https://bidsconvertr.github.io/>) and an online tutorial (<https://bidsconvertr.github.io/tutorial/>).

Results

BIDScnvertR was initially developed and validated on MRI data from the BiDirect study^{8,9}, a data set with more than 90,000 single images (for details of the study, see *Methods*). It was used to convert the DICOM data of all participants to a valid BIDS² data set. The performance of BIDScnvertR was also tested by the coauthors on different operating systems (Ubuntu 22.04 and Windows 10), different MRI vendors (Siemens and Philips) and using an independent dataset (<https://surfdrive.surf.nl/files/index.php/s/HTxdUbykBZm2cYM/download>) provided by the BIDScoin⁵ team. Furthermore, the GitHub repository (<https://github.com/bidsconvertr/bidsconvertr>) and documentation (<https://bidsconvertr.github.io/>) facilitate user interaction in case of errors or questions.

Description of the workflow. The BIDScnvertR’s workflow (Fig. 1) is condensed into a single function: `convert_to_BIDS()`. We provide a brief explanation of each step of this function below. The online documentation contains detailed information and screenshots of the workflow. Moreover, an online tutorial comprises a step-by-step walkthrough based on a sample dataset. The installation process is described in the usage notes below.

Getting started. To start the BIDSconvertR the package must be loaded (`library("bidsconvertR")`) and the (`convert_to_BIDS()`) function needs to be executed. Users can choose from existing “user_settings.R” files or create a new file from the file-selection and point-and-click pop-up boxes, which store information about file paths and input data structures. Each user interaction results in diagnostic output.

If users want to create a new file, a pop-up box appears with a user dialogue, where the user can select input and output paths with the file explorer, and is asked about the hierarchy of input folders to extract ‘session-’ and ‘subject-ID’. The required information is (1) the input path containing all subfolders with DICOMs, (2) a selection of the input folder hierarchy (choose between: session/subject or subject/session) and (3) the output folder for all temporary and BIDS files. At the end of the procedure the “user_settings.R” file is saved to the selected output folder.

The following step should enable the conversion in case of complicated subject-ID folder names into short and simple ones. The existing subject-IDs can optionally be shortened with regular expressions in the case of redundant prefixes, suffices, and strings. For instance, the template “[[:digit:]]{5}” means ‘extract any string in a filename with 5 digits’, and remove the rest of the string. Regardless of the setting, if no regular expression is added, the subject-IDs remain unchanged. Users then have the option to rename their session-IDs. Unchanged session-IDs are kept unchanged. The workflow creates background variables and file paths based on the “user_settings.R” file. If an already existing “user_settings.R” file is selected the pop-up-based user dialogue boxes are skipped.

From DICOM to NIfTI. A specified version of `dcm2niix`⁴ that worked best on our data will automatically be installed, although it is possible to install a custom version.

All DICOMs are converted twice with ‘`dcm2niix`’⁴ and then saved separately in a “`dcm2niix_converted`” and an “`identifying_information`” folder. The “`dcm2niix_converted`” folder contains NIfTI and JSON data with sensitive information removed from both files. The “`identifying_information`” folder contains only JSON sidecar files containing the sensitive privacy information (subject-ID, birthdate, age, gender, weight, height, acquisition date). Per folder, all JSON sidecar files are indexed, then the headers and all sequence parameters are extracted, merged together, and provided in one file (“`json_metadata.tsv`”) for quality control. Also, a file called “`sequence_overview.tsv`” is made with a summary of each sequence, including the amount of data for each sequence, in total and per session. This separation allows users to internally use personally identifiable information such as subject-IDs in pseudonymized datasets, while in the case of data sharing the metadata are de facto anonymized. However, additional removal of the facial structure and skull and removal of subject-IDs from filenames are still required for a fully anonymized dataset¹⁰.

From NIfTI to BIDS. The NIfTI filenames from the “`dcm2niix_converted`” folder must now be manually adapted to the BIDS specification. This is facilitated by the “sequence mapper” graphical user interface (GUI) (Fig. 2) that enables editing and live-validation of BIDS sequence IDs. The graphical user interface presents hyperlinks to the BIDS specification, which we recommend reading in its most recent version² to ensure that the chosen names are in line with the most recent changes. The following information is required: (1) the BIDS sequence ID, (2) the BIDS type (“`anat`, `dwi`, `func`, `fmap`, or `else`”), and (3) the relevance (0 = no, 1 = copy to BIDS). The entries are color-coded in red and green, with green representing valid BIDS sequence names and red indicating invalid or unedited ones. The ‘sequence mapper’ (Fig. 2) does not allow workflow progress if the field is not edited or you enter an invalid BIDS type, BIDS sequence ID, or combination of both. When the GUI is closed and each field is edited, the workflow automatically starts with the subsequent step.

The NIfTI data are copied and renamed to BIDS specification² inside the “`bids`” folder using the renaming pattern from the “sequence mapper” (Fig. 2). Each new file will be copied and renamed to the “`bids`” directory within the output folder. For each subject-ID there is an individual subfolder and within each of these subject-ID folders, the corresponding session-ID folders containing the sequence-type folders are created. The sequences can be found as NIfTI files at the lowest folder level (within the sequence-type folder). Each filename within the “`bids`” folder is specified as “`subject-id_session-id_sequence-type_sequence-id`” according to the BIDS naming convention (Fig. 1, right panel).

From here, all necessary BIDS metadata files are automatically generated. The “`CHANGES`”, “`README`” and “`dataset_description.json`” files contain placeholder content of the ‘BIDSconvertR’ package⁶, which needs to be updated by the users. It must be modified and adapted to their own data set. The sequence-agnostic BIDS JSON sidecar files (e.g., “`participants.tsv`”) are created automatically and contain the information from the anonymized JSON sidecar files in the “`bids`” folder. The metadata inside of the BIDS folder do not contain any potentially identifying information. Instead, the BIDSconvertR creates another “`participants.tsv`” in the “`identifying_information`” folder that contains the privacy-sensitive information from the JSON headers.

Diagnostics and quality control. The ‘BIDSconvertR’ features open-source dependencies: the recent version of the BIDS-validator⁷ and an embedded interactive MR image viewer based on `papaya` (<https://CRAN.R-project.org/package=papaya>, <https://neuroconductor.org/package/papayaWidget>) for visual quality control (Fig. 3). The BIDS-validator⁷ should be used to investigate potential issues with the resulting BIDS-dataset. If Docker is installed, the workflow automatically launches the latest Docker version of the BIDS-Validator in the “`bids`” folder. Otherwise, the corresponding website is launched. The final step is the automated start of the ‘Shiny BIDS’ viewer (Fig. 3). This GUI features an easy selection and navigation of the nested BIDS structure. It enables a manual inspection of the acquired data for quality control purposes with an image viewer based on `papaya` inside a graphical user interface. It can also be launched separately by using the `run_shiny_BIDS()` function and selecting a valid BIDS folder. After validation and visual quality control, users are asked to remove the “`dcm2niix_converted`” files, which are a temporary by-product of the workflow.

	sequence	BIDS_type	BIDS_sequence	relevant	valid
1	t1_mprage_sag_ipat2_1p0iso	anat	T3w	1	no
2	AAHead_Scout_32ch-head	anat	smart	0	not selected
3	cmrr_2p4iso_mb8_TR0700	func	task-reward_acq-mp8_echo-1_bold	1	yes
4	cmrr_2p4iso_mb8_TR0700_SBRef	anat	task-reward_acq-mp8_echo-1_sbref	1	no
5	cmrr_2p5iso_mb3me3_TR1500_e1	func	task-stop_acq-mp3m3_run-1_dolb	1	no
6	cmrr_2p5iso_mb3me3_TR1500_e2	func	task-stop_acq-mp3m3_run-2_bold	1	no
7	cmrr_2p5iso_mb3me3_TR1500_e3	func	task-stop_acq-mp3m3_run-3_bold	1	yes
8	cmrr_2p5iso_mb3me3_TR1500_SBRef_e1	fmap	task-stop_acq-mp3m3_run-1_sbref	1	no
9	cmrr_2p5iso_mb3me3_TR1500_SBRef_e2	func	task-stop_acq-mp3m3_run-2_sbref	1	yes
10	cmrr_2p5iso_mb3me3_TR1500_SBRef_e3	func	task-stop_acq-mp3m3_run-3_sbref	1	yes
11	field_map_2p4iso_e1	fmap	acq-2p4_magnitude1	1	yes
12	field_map_2p4iso_e2	fmap	acq-2p4_magnitude3	1	no
13	field_map_2p4iso_e2_ph	fmap	acq-2p4_phasediff	1	yes
14	field_map_2p5iso_e1	fmap	acq-2p5_magnitude1	1	yes
15	field_map_2p5iso_e2	fmap	acq-2p5_magnitude2	1	yes
16	field_map_2p5iso_e2_ph	fmap	acq-2p5_phasediff	1	yes

Fig. 2 The ‘sequence mapper’ graphical user interface. The interface starts automatically for the sequence mapping of the input sequence-IDs to BIDS sequence-IDs. The user has to edit all cells according to BIDS, click “save” and close the app. Please note the color-coding feature: Green rows indicate a valid BIDS sequence, while red rows indicate invalid BIDS sequences or cells that were not edited. Only the relevant marked sequences are copied into BIDS standard, while the white ones in the “relevant” column are skipped for conversion and validation.

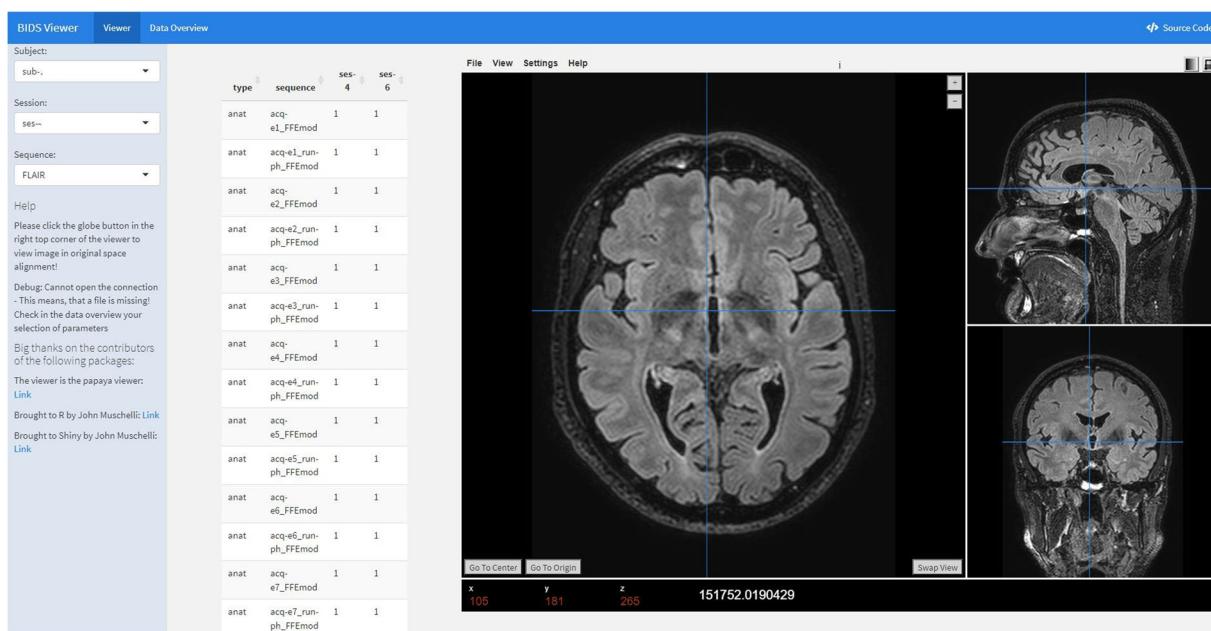


Fig. 3 The ‘BIDS viewer’ graphical user interface. The ‘BIDS viewer’ launches automatically from the ‘convert_to_BIDS()’ command when all files are converted to NIFTI and stored in the BIDS specification. The ‘papayaWidget’ viewer is used to check the visual quality control of the images. The user is able to inspect BIDS data easily without clicking through a nested folder structure. The viewer can also be used on existing BIDS datasets by giving the path as an argument to the ‘run_shiny_BIDS(“C:/path/to/BIDS_dataset/”)’ function.

Usage Notes

The following section describes the installation of the BIDSconvertR, gives some practical support for the use of regular expressions and ends with a tutorial which covers each step in the workflow of the “BIDSconvertR”⁶.

Installation processes. Users need to install a recent version of R (<https://www.R-project.org/>, tested with 4.3.1) and RStudio (<https://www.rstudio.com/>). For a Windows system, ‘rtools43’ must additionally be downloaded and installed. When using a Debian (Ubuntu) system, dependencies must be installed with the following command: `sudo apt install libcurl4-openssl-dev libssl-dev libxml2-dev libfontconfig1-dev`. For the installation of the `BIDSconvertR` the following two commands need to be entered on the R console inside RStudio: The ‘devtools’ package (<https://CRAN.R-project.org/package=devtools>) must be installed using the command `install.packages("devtools")`. To install the ‘BIDSconvertR’ from GitHub, enter `devtools::install_github("bidsconvertr/bidsconvertr")`. The package checks and installs all required libraries¹¹ automatically during the installation process and asks for updates, if necessary.

Optional feature: regular expressions. The use of regular expressions is an optional feature for the purpose of file cleaning. The following notes will deliver some basic knowledge of regular expressions by using some examples. Further information can be found in the ‘stringr’ R package¹² publication and cheat sheet, as well as in our `BIDSconvertR` documentation (https://bidsconvertr.github.io/usage_notes/#resources-for-regular-expressions).

Users can be very explicit regarding the regular expressions of the subject ID. If the subject ID consists of 5 digits, the regular expression is `"[:digit:]{5}"`. If the subject-ID consists of either ‘control_’ or ‘intervention_’ and two digits, the regular expression could look like this `"(control_|intervention_)[:digit:]{2}"`. The pipe operator ‘|’ is used as a Boolean ‘OR’ operator and can combine multiple regular expressions. Alternatively, the same expression can be generated by: `"control_[:digit:]{2}|intervention_[:digit:]{2}"`. It is recommended to use short and simple filenames. Since most of the acquisition data (e.g., date, sex) is included in the json sidecar header, it is not needed in the filename. One short and simple naming convention is one or two letters or digits defining the cohort, and sufficient digits corresponding to the intended study size (e.g., the first in a control cohort of size 100 could be named ‘c001’, or ‘i0001’ from an intervention cohort of size 1000).

If there are dispensable letters, signs, or digits in the filename, they should be added to the ‘pattern to remove’ expression. For instance, users can remove the strings ‘study’, ‘control’ and ‘patient’ with the regular expression: `"(study|control|patient)"`. To check for invalid subject-IDs, users can review the extracted subject-IDs in the diagnostic output in the terminal and the ‘dicom_paths.tsv’ file. All non-matching strings of the filename are displayed in the ‘rest’ columns of the ‘dicom_paths.tsv’ file.

Tutorial. The idea originates from the ‘BIDScoin’⁵ publication and is adapted to the ‘BIDSconvertR’. It guides users through the basic steps of the workflow on the basis of sample data. A more detailed (short) tutorial (<https://bidsconvertr.github.io/tutorial/>) and even more detailed usage notes (<https://bidsconvertr.github.io/>) that cover each step of `BIDSconvertR` can be found in the online documentation. Additionally, the tutorial dataset was used for validation of the ‘BIDSconvertR’ on an independent dataset.

Data preparation. Users can download the `BIDScoin` example data⁵ (<https://surfdrive.surf.nl/files/index.php/s/HTXdUbykBZm2cYM/download>). The data is compressed twice with gunzip (suffix: ‘gz’) and tar (suffix: ‘tar’) and needs to be unpacked. Users should then create a new folder (e.g. ‘bidscoin_example’) and copy the ‘raw’ subject folders into it.

The BIDSconvertR workflow. This are the main steps to use the ‘BIDSconvertR’ with the tutorial data:

1. Launch RStudio and use the R console from here on.
2. Load the library with the command: `library("bidsconvertr")`.
3. Use the `convert_to_BIDS()` command to launch the tool.
4. Create your own ‘user_settings.R’ file by answering the questions in the text prompts.
 - (a). Select the input folder that contains the DICOMs (e.g., ‘bidscoin_example_raw’).
 - (b). Select the ‘./subject/session/.’ order of folders.
 - (c). Select the output folder.
 - (d). Do not use the ‘subject-ID’ and ‘pattern to remove’ features (subject-IDs are valid).
5. Use the sequence mapper to rename your files according to BIDS, then save and close the app. See the online tutorial (<https://bidsconvertr.github.io/tutorial/#default-values-for-tutorial-data>) for a detailed map of the sequence names.
6. Now the data is saved in BIDS. The BIDS-Validator and the ‘Shiny BIDS viewer’ start.

Discussion

The ‘BIDSconvertR’⁶ is a free, open-source R package and the first alternative for the R community. Most BIDS converters are developed in other languages such as Python or MATLAB (<https://bids.neuroimaging.io/benefits.html>). The ‘BIDSconvertR’ extends the capabilities of BIDS² to users with a basic understanding of R and allows them to develop new features for the tool. It is simple enough that no programming experience is required to follow the tutorial (<https://bidsconvertr.github.io/tutorial/>) and use the package. In addition, the GitHub repository

(<https://github.com/bidsconvertr/bidsconvertr>) and documentation (<https://bidsconvertr.github.io/>) facilitate user interaction in case of errors or questions.

To our knowledge, it is one of the few tools (BIDScoin⁵, Horos (<https://horosproject.org/faqs/>), Osirix¹³, and BrkRaw¹⁴) (<https://bids.neuroimaging.io/benefits.html>) with a graphical user interface. In contrast to the others it simplifies the ‘sequence mapping’ with a live color-coded validation of the entered sequences and features a viewer for visual inspection of BIDS-structured MRI images.

Our tool is developed to:

- be executed continuously during the data acquisition of a study, as it checks for already converted and existing data,
- facilitate the creation of short and simple filenames,
- check and validate ongoing changes in sequence names and types,
- extract and merge all metadata information,
- detect implausibilities in sequence parameters by color-coding,
- create automatically required sequence-agnostic files like “participants.tsv”,
- validate the dataset with BIDS-Validator⁷ in Docker or the website, and
- efficiently inspect and navigate through BIDS structured MRI data with the ‘Shiny BIDS viewer’.

The ‘BIDSconvertr’ has the some limitations. It was developed only with MRI data and up to now supports only this type of data. The program requires an IDE (integrated development environment, here RStudio) and a console, which is however also a requirement for all other BIDS converters. To facilitate users that are unfamiliar with R the use of the ‘BIDSconvertr’ we linked several resources in this publication and the online documentation (<https://bidsconvertr.github.io>). A basic understanding of regular expressions is required for advanced features, like the cleaning of subject IDs. Although regular expressions are complicated to use, the documentation of the ‘BIDSconvertr’ should help the user in the application and describe the installation, workflow, and its practical use with a tutorial. Moreover, the use of regular expressions is optional and not required for the basic use of the ‘BIDSconvertr’.

Taken together, the BIDSconvertr is the first BIDS converter developed in R, which is aimed at users with a background in R (statistics or epidemiology) to get their data into BIDS without the necessity to learn Python or MATLAB and their ecosystem. In a future update, we plan to extend the ‘BIDSconvertr’⁶ with additional wrapper functions for popular processing pipelines like fMRIPrep¹⁵ or MRIQC¹⁶, allowing users to process their MRI data even more efficiently.

Methods

Study data. The data used for the development of the tool originate from the BiDirect study and is briefly described here. The BiDirect study^{8,9} is a prospective cohort study aiming to establish the bidirectional relationship between subclinical arteriosclerosis and depression. Data were acquired over a period of 12 years, including repeated MR-imaging of the brain using the same Philips scanner and sequences. The study comprised approximately 90,000 images from around 4,500 MRI sessions. The ‘BIDSconvertr’ uses ‘dcm2niix’ to convert DICOM data. Data from all major MRI vendors supported by ‘dcm2niix’ can be converted. Conversion was tested on three data sets: The BiDirect MRI data that were acquired with a Philips scanner, and two data sets from Siemens scanners, one in-house dataset and the example data provided by the BIDScoin developers⁵. A recent version of ‘dcm2niix’ (V1.0.20230411, Chris Rorden⁴) was implemented and tested for compatibility.

Software & hardware. The ‘BIDSconvertr’ package⁶ was developed with R 4.3.1 and RStudio. The packages ‘usethis’ (<https://CRAN.R-project.org/package=usethis>) and ‘devtools’ (<https://CRAN.R-project.org/package=devtools>) were used for package development. The ‘tidyverse’¹¹ functions of the packages ‘dplyr’, ‘tidyr’, ‘stringr’ were used for file renaming and restructuring. JSON files were extracted with the package ‘rjson’ (<https://CRAN.R-project.org/package=rjson>). The ‘Sequence Mapper’ GUI uses the ‘shiny’ (<https://CRAN.R-project.org/package=shiny>) and ‘DT’ (<https://CRAN.R-project.org/package=DT>) packages for the spreadsheet interaction and color-coded BIDS validation. The ‘Shiny BIDS viewer’ was made with ‘flexdashboard’ (<https://CRAN.R-project.org/package=flexdashboard>), ‘shiny’, and papaya. The conversion from DICOM to NIfTI is done by ‘dcm2niix’ (V1.0.20211006, Chris Rorden⁴), from which each feature can be implemented in the workflow. All code is hosted freely and open-source on a GitHub repository⁶.

Requirements. In general, there are a few requirements (for the data, user, and system) for the ‘BIDSconvertr’⁶. Users needs a basic understanding of the BIDS nomenclature and file management. In the case of advanced features like file cleaning, a basic understanding of regular expressions is required. Users can find useful resources on our Github README (<https://github.com/bidsconvertr/bidsconvertr>) or in the documentation (<https://bidsconvertr.github.io/>).

Input file convention. Data must be in a specific structure that corresponds to the common output of PACS (Picture Archiving and Communication System): either (1) a folder per time point or study, containing subfolders of each subject with the DICOM data (“input/session/subject”) or (2) a folder per subject, containing the time-point subfolders with the DICOM data (“input/subject/session”) (Fig. 1, left side). If there are only cross-sectional data, we recommend to create a folder named, e.g. “cross_sectional”, “baseline” or “0”, that contains all subject folders with the DICOM data. If data are structured another way, we recommend to rearrange it. To set up the hierarchical order in which the ‘BIDSconvertr’ extracts information from the data, users are required to set the option during the user dialogue either to ‘session_subject’ or ‘subject_session’. In general, we recommend

removing spaces or special text symbols in file or folder names. If the data already have the BIDS-required prefixes “sub-” or “ses-”, users do not need to change it.

Data availability

Each release is saved at Zenodo⁶, which can also be accessed via <https://doi.org/10.5281/zenodo.5878407>. We thank the developers of BIDScoin⁵ for the permission to use the sample data in the tutorial, which can be downloaded at <https://surfdrive.surf.nl/files/index.php/s/HTxdUbykBZm2cYM/download>.

Code availability

All code is hosted freely and open-source on a GitHub repository, accessible via <https://github.com/bidsconvert/bidsconvert>. The documentation is hosted on the GitHub page: <https://bidsconvert.github.io>.

Received: 3 February 2023; Accepted: 21 September 2023;

Published online: 04 October 2023

References

1. Cox, R. *et al.* A (sort of) new image data format standard: NiFTI-1. In *10th Annual Meeting of the Organization for Human Brain Mapping*, vol. 22 (2003).
2. Gorgolewski, K. J. *et al.* The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments. *Scientific Data* **3**, <https://doi.org/10.1038/sdata.2016.44> (2016).
3. Gorgolewski, K. J. *et al.* BIDS apps: Improving ease of use, accessibility, and reproducibility of neuroimaging data analysis methods. *PLoS Computational Biology* **13**, e1005209, <https://doi.org/10.1371/journal.pcbi.1005209> (2017).
4. Li, X., Morgan, P. S., Ashburner, J., Smith, J. & Rorden, C. The first step for neuroimaging data analysis: DICOM to NiFTI conversion. *Journal of Neuroscience Methods* **264**, 47–56, <https://doi.org/10.1016/j.jneumeth.2016.03.001> (2016).
5. Zwiers, M. P., Moia, S. & Oostenveld, R. Bidscoin: A user-friendly application to convert source data to brain imaging data structure. *Frontiers in Neuroinformatics* **15**, <https://doi.org/10.3389/fninf.2021.770608> (2022).
6. Wulms, N., Eppe, S. & Minnerup, H. Bidsconvert, *Zenodo*, <https://doi.org/10.5281/zenodo.8252451> (2023).
7. Gorgolewski, C. *et al.* bids-standard/bids-validator: 1.4.3, *Zenodo*, <https://doi.org/10.5281/zenodo.3688707> (2020).
8. Teismann, H. *et al.* Establishing the bidirectional relationship between depression and subclinical arteriosclerosis - rationale, design, and characteristics of the BiDirect Study. *BMC Psychiatry* **14**, <https://doi.org/10.1186/1471-244X-14-174> (2014).
9. Teuber, A. *et al.* MR imaging of the brain in large cohort studies: feasibility report of the population- and patient-based BiDirect study. *European Radiology* **27**, 231–238, <https://doi.org/10.1007/s00330-016-4303-9> (2017).
10. Eke, D. *et al.* Pseudonymisation of neuroimages and data protection: Increasing access to data while retaining scientific utility. *Neuroimage: Reports* **1**, 100053, <https://doi.org/10.1016/j.ynirp.2021.100053> (2021).
11. Wickham, H. *et al.* Welcome to the Tidyverse. *Journal of Open Source Software* **4**, 1686, <https://doi.org/10.21105/joss.01686> (2019).
12. Wickham, H. stringr: Simple, consistent wrappers for common string operations. <https://doi.org/10.6084/m9.figshare.853801>. R package version 1.4.0 (2019).
13. Rosset, A., Spadola, L. & Ratib, O. OsiriX: An Open-Source Software for Navigating in Multidimensional DICOM Images. *Journal of Digital Imaging* **17**, 205–216, <https://doi.org/10.1007/s10278-004-1014-6> (2004).
14. Lee, S.-H., Ban, W. & Shih, Y.-YI. Brkraw/bruker: Brkraw v0.3.3, *Zenodo*, <https://doi.org/10.5281/zenodo.3877179> (2020).
15. Esteban, O. *et al.* fMRIPrep: a robust preprocessing pipeline for functional MRI. *Nature Methods* **16**, 111–116, <https://doi.org/10.1038/s41592-018-0235-4> (2019).
16. Esteban, O. *et al.* MRIQC: Advancing the automatic prediction of image quality in MRI from unseen sites. *PLoS One* **12**, e0184661, <https://doi.org/10.1371/journal.pone.0184661> (2017).

Acknowledgements

We thank everyone who was involved in the BiDirect study. Most importantly, we thank the BiDirect participants who made this research possible. We appreciate the BIDScoin⁵ team’s permission to use their sample data. The BiDirect Study is funded by the Federal Ministry of Education and Research, Germany (BMBF, grants 01ER0816, 01ER1205 and 01ER1506).

Author contributions

N.W.: developed the concept, visualisation, and code; drafted the manuscript. S.E.: substantial contributions to the design of the code; revision of the manuscript for important intellectual content. M.D.: substantial contributions to testing and identifying errors of the ‘BIDSconvertR’ on independent data from different MRI vendors; revision of the manuscript for important intellectual content. A.S.: substantial contributions to testing and identifying errors of the documentation; revision of the manuscript for important intellectual content. N.B.: substantial contributions to testing and identifying errors of the tutorial; revision of the manuscript for important intellectual content. K.B.: substantial contributions to the acquisition of data; revision of the manuscript for important intellectual content. B.S.: substantial contributions to the conception and functionality of the code; revision of the manuscript for important intellectual content. H.M.: substantial contributions to the conception and acquisition of data; drafting and revision of the manuscript for important intellectual content.

Funding

Open Access funding enabled and organized by Projekt DEAL.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to N.W.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023