SOVABIDS

A python backend package for the conversion of EEG datasets to the BIDS standard, with a focus on automation, reproducibility and interoperability.

Mantilla Ramos Y [1–5], Hoyos Madera B [1–5], Bollmann S [2], Narayanan A [2–3], White D [4], Johnstone T [3–4], Civier O [3–4]

1 Grupo Neuropsicología y Conducta, Universidad de Antioquia, School of Medicine, Medellín Antioquia, Colombia

2 The University of Queensland, Brisbane Queensland, Australia

3 Australian National Imaging Facility, Australia

4 Swinburne Neuroimaging, Swinburne University of Technology, Melbourne Victoria, Australia

5 Semillero de Investigación NeuroCo, Universidad de Antioquia, School of Medicine & School of Engineering, Medellín Antioquia, Colombia

CONTACT
yjmantilla@gmail.com
sovabids.readthedocs.io
github.com/yjmantilla/sovabids

INTRODUCTION

- The Brain Imaging Data Structure (BIDS) is a standard for neuroimaging datasets that helps with data sharing and reusability; it has gained popularity within the electroencephalography (EEG) community.
- Converting raw data to BIDS is not technically difficult but involves substantial effort if done manually.
- Software is available to assist the conversion, but it requires either basic programming skills or detailed user input.
- This creates a barrier to the adoption of BIDS, thus impeding the goals of reproducibility and data reuse in the EEG community.

OBJECTIVE

Develop a software that helps the adoption of BIDS within the EEG community by adapting to its needs.

EEG TO BIDS CONVERSION

METHODOLOGY

The following set of constraints were identified given the context of the problem:

Maintainability and open-source development, given the community focus of the BIDS standard.

No programming skills needed, to make the usage easier for non-technical users.

Handling of outliers, as files with minor differing details are a common problem in EEG.

Reproducible conversion, which is important for research reproducibility.

Interoperability with other software, to make sovabids useful for other developers and embed it in multiple GUIs and platforms.

Minimal user-input, to make the conversion as automated as possible.
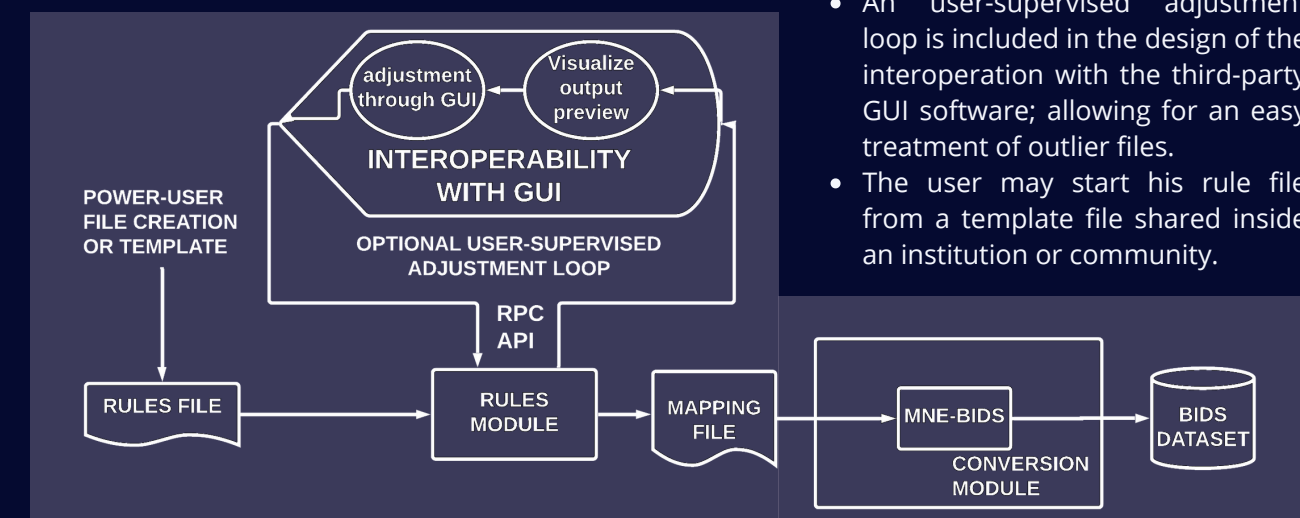
Broad support of formats, for a greater adoption among the community.

Do not reinvent the wheel; use available software that can be adapted to our requirements.

With these requirements in mind, the subsequent design decisions were made:

- Focus on the development of a backend rather than a GUI as it is easier to maintain in the long-run.
- Git version control.
- CI automated testing through Github Actions to have fast feedback on the plausibility of new changes.
- Code-Coverage to improve testing and increase the overall code quality.
- Automated Documentation through Sphinx, allowing for faster and easier documentation.

- Include usage examples for the user through the Sphinx Gallery Extension.
- Use two human-readable YAML configuration files: the **Rules File** and the **Mappings File**.

The Rules File
Encodes the general conversion rules for a dataset; in this way it helps minimizing future user input.

Heuristics
Several are implemented to extract information from the file paths. This allows to take advantage of the structured file-system patterns used in EEG research.

Rules File → *Mappings File*

From a **Rules File**, a mapping for each file in the dataset can be generated and saved in the **Mappings File**.

The Mappings File
Encodes the specific conversion rules for an individual EEG file. It is useful for outlier files like those common in EEG. This mapping can be customized by the user in a number of ways and it fully defines a conversion process.

- Provide a Web API for interoperability.
- Given the action-oriented methods that naturally adjust to a conversion process, we use a RPC protocol; specifically JSON-RPC as it is easier to understand and is less verbose (in contrast to XML).

- Include the parameters needed to fully replicate the conversion process in the configuration files, along with the provenance of the information used in the conversion for later revision.

- Use MNE as it supports the reading of most EEG file formats.

- Use MNE-BIDS, as it is already capable of performing conversions using detailed inputs from a technical user. Nevertheless, it is limited in terms of generalization capability.

ARCHITECTURE

An open-source python package comprising two main modules was developed:

- A rules module to encode and interpret the rules of the conversion.
- A module interfacing with MNE-BIDS to execute the conversion

- An user-supervised adjustment loop is included in the design of the interoperation with the third-party GUI software; allowing for an easy treatment of outlier files.
- The user may start his rule file from a template file shared inside an institution or community.

REFERENCES

- Pernet, C.R., Appelhoff, S., Gorgolewski, K.J. et al. EEG-BIDS, an extension to the brain imaging data structure for electroencephalography. Sci Data 6, 103 (2019). https://doi.org/10.1038/s41597-019-0104-8
- Alexandre Gramfort, Martin Luessi, Eric Larson et al. MEG and EEG data analysis with MNE-Python. Frontiers in Neuroscience, 7(267):1–13, 2013. doi:10.3389/fnins.2013.00267.
- Appelhoff, S., Sanderson, M., Brooks, T. et al. (2019). MNE-BIDS: Organizing electrophysiological data into the BIDS format and facilitating their analysis. Journal of Open Source Software, 4:1896. DOI: 10.21105/joss.01896

RESULTS

- The implemented solution enables the reproducible semi-automatic conversion of EEG datasets through the use of human-readable configuration files. It is also interoperable with external frontends, making it more accessible for less-technical users and also faster than fully manual-input solutions.
- The extensive documentation and CI pipelines facilitate the maintenance of the open-source code by the community.
- Interoperability was tested using a minimal demonstrative GUI developed in Flask for this specific purpose (see the following link: sovabids.readthedocs.io/en/latest/auto_examples/gui_example.html).

CONCLUSIONS

- An interoperable, reproducible and maintainable EEG-to-BIDS conversion software was developed.
- Future work will focus on leveraging diverse metadata files from different EEG vendors through the implementation of extensible heuristics, allowing for greater automation and a broader support of formats.

ACKNOWLEDGMENTS