GitMA Poster

Meister, Malte

meister@linglit.tu-darmstadt.de
Technische Universität Darmstadt, Germany

Gerstorfer, Dominik

dominik.gerstorfer@tu-darmstadt.de
Technische Universität Darmstadt, Germany

Schumacher, Mareike Katharina

schumacher@linglit.tu-darmstadt.de
Technische Universität Darmstadt, Germany

Gius, Evelyn

evelyn.gius@tu-darmstadt.de
Technische Universität Darmstadt, Germany

To remember something does not merely mean to save it, but also to retrieve it and process it further. Only when processed further does the memory gain meaning. This observation applies *a fortiori* to technical storage systems. The benefit of a piece of software is, especially in the Digital Humanities, determined by the possibilities to export, convert, archive and further process with other tools or systems the generated data. With this poster we will present new possibilities (including a variety of visualizations and the calculation of agreement scores) that exist to retrieve and further process annotation data from CATMA (Gius et al. 2022).

CATMA (Computer Assisted Text Markup and Analysis) is a collaborative text annotation and analysis platform which is well established in the Digital Humanities and is actively used by many projects [1,2]. Exports of annotations, especially in the XML-TEI format, have already been an important part of the CATMA software since version 3 (Petris / Meister 2016; Petris 2017). However, up until and including CATMA 5, data could only be accessed via the graphical user interface (GUI). Since version 6.0 the data that are generated by users are stored and version controlled in a Git-based backend.

The exact details about the data structures are documented on the CATMA website (Petris 2020): Each document, each annotation collection, as well as each tagset is separately represented in the backend. Especially important for the further processing of annotation data are the separate pieces of information that represent distinct annotations:

- a reference to the corresponding document
- the exact placement of the annotated text segment (as so-called start and end-offsets, which reference the character positions within the text)
- a reference to the tag (the annotation category) that was utilized as well as the tagset (a named collection of tags) from which it stems
- possibly properties (pre-defined extending attributes) and their values
- author of the annotation
- timestamp of the annotation

Since the introduction of the Git-based backend users can access their own data, as well as those that have been shared with them, in the form of Git repositories. These function as an application programming interface (API) for the retrieval of annotation data, which can be downloaded to the local computer or processed further using other tools.

In addition, a Python library (Vauth et al. 2022) which enables easy access to the data within the Git repositories directly from the coding environment was developed at the department of Digital Philology at the Technical University of Darmstadt. It makes it possible to further process annotations using established and popular Python data science tools, for example as a Pandas DataFrame. Among other things, the Python library allows for the calculation of Inter-Annotator Agreement or the creation of visualizations, such as for tracking annotation progress and further exploration of the annotations. In this way not only the evaluation of annotations, but also rapid identification and immediate correction of annotation errors is made possible.

The overall objective of the Git Access is to make CATMA data directly accessible, so that users aren't necessarily limited by those functionalities that already exist in CATMA. Thereby the workflow encompassing annotation, evaluation of annotations and revision of annotations can be accelerated significantly and easily adapted. This is especially relevant for users that are concerned with the organization and evaluation of annotations — for example in the context of larger collaborative research projects or in lecture settings.

We will present this workflow in detail with our poster. Thus the poster is also meant to serve as a kind of instruction manual for the use of the CATMA Git Access and to present best practices. The following steps will be covered:

1. Prerequisites for accessing the CATMA GitLab API
2. Installation of the GitMA Python library (or rather a Docker image which covers all requirements)
3. Cloning the repositories
4. Accessing the data using Python
5. Examples of annotation exploration and evaluation

[1] For example, CATMA appears in the Awesome Digital Humanities list at https://dh-tech.github.io/awesome-digital-humanities/, the TAPoR Tools-Index at https://tapor.ca/tools/1469 and in "Digitale Werkzeuge zur textbasierten Annotation, Korpusanalyse und Netzwerkanalyse in den Geisteswissenschaften" (Frey-Endres / Simon 2021).

[2] Of course CATMA is not the only tool of its kind. You can find many other similar and not-so-similar tools and a wide range of other resources in a number of curated lists and indices, for example: the Awesome Digital Humanities list at https://dh-tech.github.io/awesome-digital-humanities/ (which itself links to other such collections under the heading "Other Resources"), theTAPoR Tools-Index at https://tapor.ca/, the annual results of the Digital Humanities Awards at http://dhawards.org/dhawards2022/results/(most recent at time of writing) and "Digitale Werkzeuge zur textbasierten Annotation, Korpusanalyse und Netzwerkanalyse in den Geisteswissenschaften" (Frey-Endres / Simon 2021).

Bibliography

Frey-Endres, Marcel / Simon, Tobias (2021): "Digitale Werkzeuge zur textbasierten Annotation, Korpusanalyse und Netzwerkanalyse in den Geisteswissenschaften", in: *Digital Philology | Working Papers in Digital Philology 02|2021*. Darmstadt: TU-Prints. https://tuprints.ulb.tu-darmstadt.de/17850/1/Digital_Philology__Working_Papers_in_Digital_Philology_vol002.pdf [18.04.2023].

Gius, Evelyn / Meister, Jan Christoph / Meister, Malte / Petris, Marco / Bruck, Christian / Jacke, Janina / Schumacher, Mareike / Gerstorfer, Dominik / Flüh, Marie / Horstmann, Jan (2022): *CATMA 6 (Version 6.5)*. https://catma.de/ [18.04.2023]. DOI: 10.5281/zenodo.1470118.

Petris, Marco (2017): “TEI Export Format”, in: *CATMA*. https://catma.de/documentation/access-your-project-data/tei-export-format/ [18.04.2023].

Petris, Marco (2020): “Git Access”, in: *CATMA*. https://catma.de/documentation/access-your-project-data/git-access/ [18.04.2023].

Petris, Marco / Meister, Malte (2016): “Technology and Versions”, in: *CATMA*. https://catma.de/documentation/technology-and-versions/ [18.04.2023].

Vauth, Michael / Meister, Malte / Hatzel, Hans Ole / Gerstorfer, Dominik / Gius, Evelyn (2022): *GitMA (Version 1.4.9)*. https://github.com/forTEXT/gitma [18.04.2023]. DOI: 10.5281/zenodo.5669221.