Continuous Annotation Quality Control, Support for Hierarchically Structured Label Sets and Long-Segment Annotation with WAT-SL 2.0

Christina Lohr,⋆ Johannes Kiesel,◦ Stephanie Luther,⋆ Johannes Hellrich,⋆ Benno Stein,◦ Udo Hahn⋆

⋆ Jena University Language & Information Engineering Lab (JULIE Lab),
Friedrich-Schiller-Universität Jena, Jena, Germany, <firstname.lastname>@uni-jena.de
◦ Faculty of Media,
Bauhaus-Universität Weimar, Weimar, Germany, <firstname.lastname>@uni-weimar.de

Abstract

Today's widely used annotation tools were designed for annotating typically short textual mentions of entities or relations, making their interface cumbersome to use for long(er) stretches of text, e.g, sentences running over several lines in a document. They also lack systematic support for hierarchically structured labels, i.e., one label being conceptually more general than another (e.g., *anamnesis* in relation to *family anamnesis*). Moreover, as a more fundamental shortcoming of today's tools, they provide no continuous quality control mechanisms for the annotation process, an essential feature to intrinsically support iterative cycles in the development of annotation guidelines. We alleviated these problems by developing WAT-SL 2.0, an open-source web-based annotation tool for long-segment labeling, hierarchically structured label sets and built-ins for quality control.

1 Introduction

In the course of large-scale annotation campaigns on medical full-text corpora, we encountered several shortcomings of the current generation of annotation tools. *Labeling long-spanning text segments* (e.g., entire sentences or even paragraphs) is a major issue here that is only insufficiently supported by general purpose open-source annotation tools (Müller and Strube, 2006; Stenetorp et al., 2012; Bontcheva et al., 2013; Rak et al., 2014; Yimam et al., 2014) which typically aim at annotating (much) shorter text spans for entities and relations. This is especially troublesome given the increasing availability of full texts and even books as input for annotation projects.

With annotation schemes becoming more and more conceptually structured, we also faced problems with the lack of systematic support for *hierarchically structured tag labels* where one label is semantically more general than another (e.g., the general tag *anamnesis* in relation to more specific ones like *family anamnesis*).

Finally, and this point addresses a more general design desideratum, we encountered a substantial lack of continuous *quality control* mechanisms in the majority of annotation tools (the WASA tool (AlGhamdi and Diab, 2018) is one of the rare exceptions and shares several design goals with WAT-SL 2.0). This shortcoming requires annotation project managers to reach for external tools for statistical evaluation. As a consequence, shifting back and forth between annotation and evaluation environments slows down the overall progress of the entire annotation project and hampers iterative refinement of annotation guidelines. Yet, a close technical coupling of such test-development cycles within *one* integrated platform is a particularly fruitful strategy in complex annotation campaigns.

As a remedy for these problems, we here present WAT-SL 2.0, an open source *web-based annotation tool for segment labeling*, hierarchically structured label sets and built-ins for quality control that is available under the MIT License.[1] It provides a live view on each annotator's progress on assigned documents and document sets and features Krippendorff's α (Krippendorff, 1970) for agreement statistics. WAT-SL 2.0 is based on WAT-SL, the Web Annotation Tool for Segment Labeling (Kiesel et al., 2017).

WAT-SL 2.0 was successfully employed in an on-going annotation project comprising approxi-

[1] https://github.com/webis-de/wat

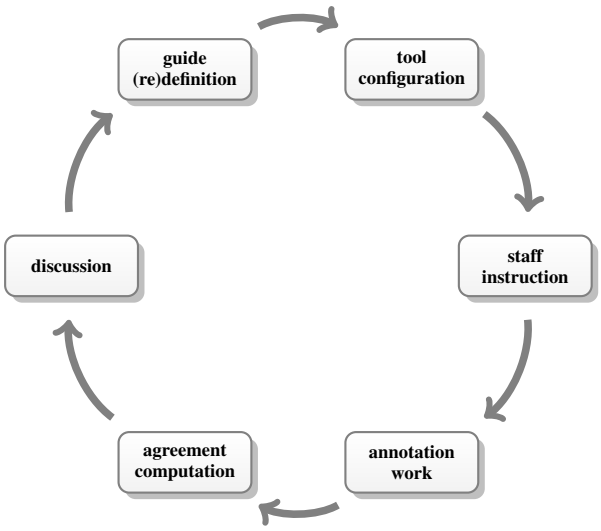

Figure 1: Enhanced annotation life cycle model (based on Pustejovsky and Stubbs (2012)'s four-step model).

mately 1K German clinical reports (Lohr et al., 2018; Hahn et al., 2018). The segment labeling subtask (see Section 4) of this project could not have been accomplished without WAT-SL 2.0's novel features and its new interface functionality.

Annotating large corpora typically requires multiple iterations to refine annotation guidelines and train annotators. This can be illustrated by the enhanced annotation life cycle model in Figure 1. Given a data collection, first an annotation guide has to be (re-)defined. Next annotation tools are configured to support the proper application of these guidelines and annotation staff is trained on them. After that the main annotation process is started and its outcome is evaluated. Finally, the overall process should be discussed by the annotation team and future iterations can be run with changed annotation guidelines and retraining, thus reflecting the experience from earlier cycles.

2 Basic Design of the Annotation Tool

WAT-SL 2.0's basic design follows WAT-SL 1.0 in providing a highly customizable and extensible interface for the annotation of full texts. It is implemented with a JAVA back-end and a Web-based front-end making it highly compatible with different environments and easy to customize. Plain text files are used as input, each line containing one segment for labeling. Results as well as logging information (e.g., time stamps) are stored in key-value files. These easy to process formats made WAT-SL 1.0 already well-suited for large-scale annotation projects and were further extended by us as described in Section 3.3.

The user interface provides annotators not only with a single document view for on-going annotation, but also with an overview page showing their upcoming and finalized tasks, as well as their progress so far—annotators in our clinical annotation project (see Section 4) found this feature particularly favorable to increase their motivation. Last but not least, WAT-SL 2.0 provides two novel administrative views (see next section) showing the progress of all annotators, as well as their agreement on specific documents.

3 Novel Features of the Annotation Tool

WAT-SL 2.0 has more advanced features—both for supporting the annotation process, as well as for servicing quality control concerns—than WAT-SL, its predecessor described by Kiesel et al. (2017), and many other tools widely used in the annotation community, BRAT (Stenetorp et al., 2012), in particular. Its features support both annotators and project managers to allow for faster and easier annotation and monitoring.

3.1 Advanced Annotation Functionality

WAT-SL 2.0 was extended with several features to allow for the large-scale annotation of documents with longer text passages using a large number of different labels.

We added support for hierarchically structured label sets for conceptually more adequate modeling of complex domains, such as clinical activities. Figure 2 shows the drop-down menu used to either directly select a label without sublabels (e.g., *preamble*) or a label with sublabels, such as the selected *anamnesis* tag. Selecting a label with sublabels prompts another drop-down menu to appear providing access to all the sublabels of the selected superlabel (e.g., selecting the superlabel *anamnesis* yields access to its conceptually more specialized sublabels *patient anamnesis* and *family anamnesis*).

Although this feature slightly increases interface complexity for the users, it considerably reduces the visual effort to pinpoint labels in the menu. Moreover, it also avoids excessively long drop-down menus that extend beyond the bottom border of the browser viewport. We successfully applied this design in a task with up to 21 labels in a preliminary annotation iteration and 18 labels (including seven hierarchical sublabels) in the final annotation project (see Section 4).

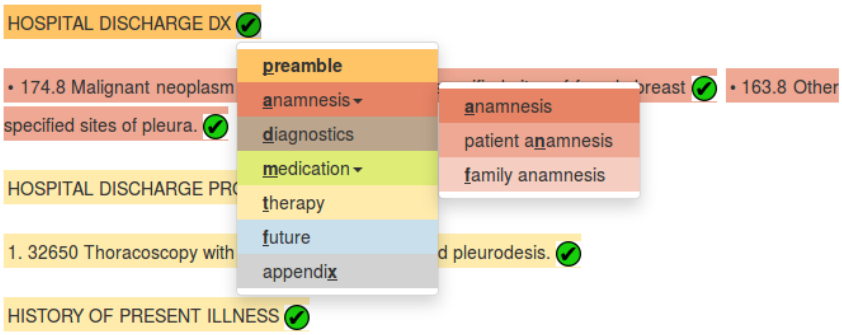

Figure 2: Sublabels of *anamnesis* tag in a secondary drop-down menu shown when the user clicks on the superlabel *anamnesis*; the bold and underlined letters display the shortcuts of the labels.

Following annotator feedback during early iterations of our annotation project, we also introduced keyboard shortcuts for each label, thus increasing both annotation speed and convenience of use. The shortcut key for each label is defined as part of WAT-SL 2.0's configuration file. The annotation drop-down menu provides both a mouse-based option to perform annotations, as well as typographic indicators for the relevant shortcuts as part of individual label names. To further support keyboard-based operation, we also introduced another shortcut (bound to the tabulator key) to select the next segment for annotation. We found these shortcuts to speed up the entire annotation process considerably, especially when labels change infrequently in long stretches of text (see Section 4 for details).

3.2 Annotation Monitoring & Quality Control

We also added advanced features for continuous progress monitoring and quality control. A single administrative interface (see Figure 3 (a)) provides annotator-specific progress reports, i.e., task and segment completion, as well as time spent on each task, and an option to take the role of any single annotator. The latter feature allows inspection and correction (logging provided for correct attribution) of individual segment annotations. We also provide a task-specific progress report for each annotator (see Figure 3 (b)) to support more fine-grained monitoring.

Finally, we added continuous quality monitoring as a task-oriented, yet annotator-agnostic view. As shown in Figure 4, this feature provides data on the progress of each annotation task and inter-annotator agreement values of the tasks completed by all annotators. Krippendorff's α (Krippendorff, 1970) is the metric of choice in WAT-SL 2.0 for measuring the chance-corrected overlap in annotation decisions. Following Artstein and Poesio (2008), we prefer it over a range of alternative measures, like Cohen's κ (Cohen, 1960), which are overly sensitive to individual annotators' decisions when modeling chance agreement.

Based on such kind of statistical evidence, continuous quality monitoring allows annotation project managers to assess the difficulty of tasks, allowing for a swift refinement of annotation guidelines. This feature was implemented by calculating coincidence matrices for each task with DKPRO AGREEMENT (Meyer et al., 2014).

3.3 Export format

WAT-SL 2.0 also provides extended export functions to increase interoperability. In addition to WAT-SL 1.0's key-value export format, we also provide CSV files well-suited as input for machine learning tools. Furthermore, we provide an export option compatible with the widely used BRAT tool, i.e., ANN files similar to the format used in the BioNLP Shared Task.[2] This increased interoperability was vital for our multi-level annotation project described in the next section.

4 Clinical Annotation Project

We employed WAT-SL 2.0 in a large-scale annotation project aiming at the creation of a reference corpus of German clinical language (Hahn et al., 2018). We annotated approximately 1K clinical documents with around 170K text segments (Lohr et al., 2018). This project covers multiple linguistic layers in addition to text segments, such as named entities (e.g., medications, diseases, etc.) and their relations (e.g., drug-drug interactions, temporal relations between clinical episodes, etc.).

[2] http://2011.bionlp-st.org/home/file-formats

Annotator	Tasks		Segments		Time	Login (new window)
Alice Carroll (alice)	5/6	83%	144/164	88%	0:14:42	Login as Alice Carroll
Bob Builder (bob)	5/6	83%	144/164	88%	0:14:16	Login as Bob Builder
Charlie Brown (charlie)	4/6	67%	82/164	50%	0:08:24	Login as Charlie Brown
Dorothy Gale (dorothy)	5/6	83%	144/164	88%	0:13:24	Login as Dorothy Gale
Frodo Baggins (frodo)	5/6	83%	144/164	88%	0:13:40	Login as Frodo Baggins

(a)

Annotator: **Bob Builder (bob)**

Task	Segments		Time
task-01	21/21	(100.0%)	0:05:08
task-02	64/64	(100.0%)	0:05:06
task-03	26/26	(100.0%)	0:01:54
task-04	18/18	(100.0%)	0:01:28
task-05	15/15	(100.0%)	0:00:40
task-06	0/20	(0.0%)	--

(b)

Figure 3: Project manager's view of progress tracking—(a) by annotator and (b) by task for a single annotator. Columns show the progress in relation to tasks and segments, the time spent and a button to log in as individual annotator (for corrections).

Task	Segments		Annotators		Kripp. Alpha
task-01	168/168	(100%)	8/8	(100%)	**0.928**
task-02	450/512	(87%)	7/8	(87%)	n.n.
task-03	208/208	(100%)	8/8	(100%)	**1.0**
task-04	144/144	(100%)	8/8	(100%)	**0.741**
task-05	120/120	(100%)	8/8	(100%)	**0.833**
task-06	0/160	(0%)	0/8	(0%)	n.n.

Figure 4: Progress monitoring by tasks and display of Inter-Annotator-Agreement. Columns show the progress in tasks, segments and Krippendorff's α.

Section annotations were performed by up to eight medical students supervised by two annotation managers with a computer science background and further advised by clinical doctors. We iteratively developed and refined guidelines for annotating segments in accordance with existing clinical requirements and standards (see Table 1). We experimented with up to 21 different labels during early exploratory iterations, but finally decided on 18 labels (including 7 hierarchical sublabels) for the final annotation round.

The first three iterations were run with the original version of WAT-SL. However, based on consistent feedback from our annotators, a desire for continuous quality control and faster iterations became obvious. Hence, we decided to implement WAT-SL 2.0. Our interface improvements contributed—probably together with a general training effect—to halving average annotation times per document from approximately 9 minutes to less than 4 minutes. Overall, our improvements clearly increased the general usability of WAT-SL and were vital for the success of our project by increasing annotation quality (effectiveness) and speed (efficiency).

Iteration	Doc.	Labels	Ø min / doc	WAT-SL
1	240	6	7:45	1.0
2	400	7	7:47	1.0
3	392	21	9:17	1.0
4	400	19	4:46	2.0
Final	1406	18	3:16	2.0

Table 1: Details for each annotation iteration. The total number of documents is inflated due to multiple annotations (by eight annotators) for agreement calculation.

5 Conclusions

We here presented WAT-SL 2.0, a Web-based tool for annotating long texts with (hierarchical) segment labels and built-in facilities for quality measurement. It provides annotators with individual progress overviews, label shortcuts and hierarchically structured label sets which help increase motivation, quality and speed for task completion. Alternative annotation tools (e.g., BRAT (Stenetorp et al., 2012) as a main representative) are mostly ill-suited for applying a large amount of labels to text segments, as they use mouse-based selection of arbitrary text spans (more suited for short-spanning entities and relations) and are thus prone to miss-clicks or lack support for both hierarchical and larger numbers of labels to select.

WAT-SL 2.0's unique elaborated monitoring device includes means for in-depth logging, annotation complexity analysis and continuous quality control. These features allow project managers to make more informed decisions when updating annotation guidelines or evaluating annotators.

We successfully employed WAT-SL 2.0 for the annotation of roughly 1K clinical reports incorporating more than 20 different labels. Furthermore, WAT-SL 2.0 is highly customizable and well-suited for non-clinical annotation tasks as well.

Acknowledgements

This work was partially supported by the Deutsche Forschungsgemeinschaft (DFG) under grants HA 2097/8-1 within the STAKI^{2}B^2 project (Semantic Text Analysis for Quality-controlled Extraction of Clinical Phenotype Information within the Framework of Healthcare Integrated Biobanking) and the German Federal Ministry of Education and Research (BMBF) within the SMITH project under grant 01ZZ1803G.

References

Fahad AlGhamdi and Mona T. Diab. 2018. WASA: a Web application for sequence annotation. In *LREC 2018 — Proceedings of the 11th International Conference on Language Resources and Evaluation. Miyazaki, Japan, May 7-12, 2018*, pages 1073–1077, Paris. European Language Resources Association (ELRA).

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Kalina Bontcheva, Hamish Cunningham, Ian Roberts, Angus Roberts, Valentin Tablan, Niraj Aswani, and Genevieve Gorrell. 2013. GATE TEAMWARE: a Web-based collaborative text annotation framework. *Language Resources and Evaluation*, 47(4):1007–1029.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Udo Hahn, Franz Matthies, Christina Lohr, and Markus Löffler. 2018. 3000PA: towards a national reference corpus of German clinical language. In *MIE 2018 — Proceedings of the 29th Conference on Medical Informatics in Europe. Building Continents of Knowledge in Oceans of Data: The Future of Co-Created eHealth. Gothenburg, Sweden, 24-26 April 2018*, number 247 in Studies in Health Technology and Informatics, pages 26–30, Amsterdam, Berlin, Washington, D.C. IOS Press.

Johannes Kiesel, Henning Wachsmuth, Khalid Al-Khatib, and Benno Stein. 2017. WAT-SL: a customizable Web annotation tool for segment labeling. In *EACL 2017 — Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Software Demonstrations. Valencia, Spain, April 5-6, 2017*, pages 13–16, Stroudsburg/PA. Association for Computational Linguistics (ACL).

Klaus Krippendorff. 1970. Estimating the reliability, systematic error and random error of interval data. *Educational and Psychological Measurement*, 30(1):61–70.

Christina Lohr, Stephanie Luther, Franz Matthies, Luise Modersohn, Danny Ammon, Kutaiba Saleh, Andreas Henkel, Michael Kiehntopf, and Udo Hahn. 2018. CDA-compliant section annotation of German-language discharge summaries: guideline development, annotation campaign, section classification. In *AMIA 2018 — Proceedings of the 2018 Annual Symposium of the American Medical Informatics Association. Data, Technology, and Innovation for Better Health. San Francisco, California, USA, November 3-7, 2018*, pages 770–779.

Christian M. Meyer, Margot Mieskes, Christian Stab, and Iryna Gurevych. 2014. DKPRO AGREEMENT: an open-source JAVA library for measuring inter-rater agreement. In *COLING 2014 — Proceedings of the 25th International Conference on Computational Linguistics: System Demonstrations. Dublin, Ireland, August 25-26, 2014*, pages 105–109. International Committee on Computational Linguistics (ICCL).

Christoph Müller and Michael Strube. 2006. Multi-level annotation of linguistic data with MMAX2. In Sabine Braun, Kurt Kohn, and Joybrato Mukherjee, editors, *Corpus Technology and Language Pedagogy. New Resources, New Tools, New Methods*, number 3 in english corpus linguistics, pages 197–214. Peter Lang, Frankfurt a.M., Germany.

James D. Pustejovsky and Amber Stubbs. 2012. *Natural Language Annotation for Machine Learning. A Guide to Corpus-Building for Applications*. O'Reilly Media, Sebastopol/CA.

Rafal Rak, Jacob Carter, Andrew D. Rowley, Riza Theresa Batista-Navarro, and Sophia Ananiadou. 2014. Interoperability and customisation of annotation schemata in ARGO. In *LREC 2014 — Proceedings of the 9th International Conference on Language Resources and Evaluation. Reykjavik, Iceland, May 26-31, 2014*, pages 3837–3842. European Language Resources Association (ELRA).

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. BRAT: a Web-based tool for NLP-assisted text annotation. In *EACL 2012 — Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics: Demonstrations. Avignon, France, April 25-26, 2012*, pages 102–107, Stroudsburg/PA. Association for Computational Linguistics (ACL).

Seid Muhie Yimam, Richard Eckart de Castilho, Iryna Gurevych, and Chris Biemann. 2014. Automatic annotation suggestions and custom annotation layers in WEBANNO. In *ACL 2014 — Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations. Baltimore, Maryland, USA, June 23-24, 2014*, pages 91–96, Stroudsburg/PA. Association for Computational Linguistics (ACL).