CARE: Collaborative AI-Assisted Reading Environment

Dennis Zyska*, **Nils Dycke***, **Jan Buchmann, Ilia Kuznetsov, Iryna Gurevych**
Ubiquitous Knowledge Processing Lab (UKP Lab)
Department of Computer Science and Hessian Center for AI (hessian.AI)
Technical University of Darmstadt
ukp.informatik.tu-darmstadt.de

Abstract

Recent years have seen impressive progress in AI-assisted writing, yet the developments in AI-assisted *reading* are lacking. We propose *inline commentary* as a natural vehicle for AI-based reading assistance, and present CARE: the first open integrated platform for the study of inline commentary and reading. CARE facilitates data collection for inline commentaries in a commonplace collaborative reading environment, and provides a framework for enhancing reading with NLP-based assistance, such as text classification, generation or question answering. The extensible behavioral logging allows unique insights into the reading and commenting behavior, and flexible configuration makes the platform easy to deploy in new scenarios. To evaluate CARE in action, we apply the platform in a user study dedicated to scholarly peer review. CARE facilitates the data collection and study of inline commentary in NLP, extrinsic evaluation of NLP assistance, and application prototyping. We invite the community to explore and build upon the open source implementation of CARE[1].

1 Introduction

Individual and collaborative text work is at the core of many human activities, including education, business, and research. Yet, reading text is difficult and takes considerable effort, especially for long and domain specific texts that require expert knowledge. While past years have seen great progress in analyzing and generating text with the help of AI – culminating in strong generative models like GPT-3 (Brown et al., 2020) and ChatGPT (Ouyang et al., 2022)[2] – the progress in applications of AI to reading and collaborative text work is yet to match these achievements. The ability of modern generative models to create natural-sounding but factually flawed outputs (Ji et al., 2022) stresses the need for supporting humans in critical text assessment.

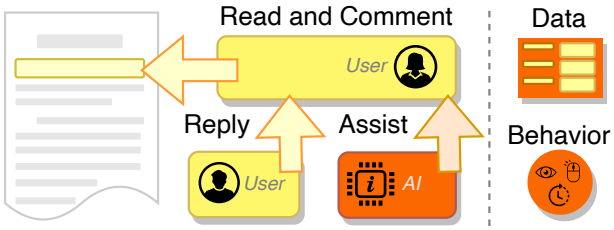

Figure 1: CARE allows users to collaboratively read and discuss texts, provides a generic interface for AI-based reading assistance, and collects research-ready textual and behavioral data.

Humans use annotations to read and collaborate over text, from hand-written print-out notes to highlights in collaborative writing platforms. This makes in-text annotations – *inline commentaries* – a promising vehicle for AI-based reading assistance. For example, an AI assistant could automatically classify the user's commentaries, or verify and provide additional information on the highlighted passages. Yet, the lack of data and key insights limits the NLP progress in this area: from the foundational perspective, we lack knowledge about the language of inline commentaries, as most of this data is not openly available for research. From the applied perspective, little is known about the hands-on interactions between humans and texts, how they translate into NLP tasks, and how the impact of NLP-based assistance on text comprehension can be measured. While ethical, controlled data collection has been receiving increasing attention in the past years (Stangier et al., 2022), data collection tools for inline commentary are missing, and so are the tools for applying and evaluating NLP models within a natural reading environment.

To address these limitations, we introduce CARE: a **C**ollaborative AI-**A**ssisted **R**eading **E**nvironment, where users can jointly produce in-

*These authors contributed equally to this work
[1] https://github.com/UKPLab/CARE
[2] https://openai.com/blog/chatgpt

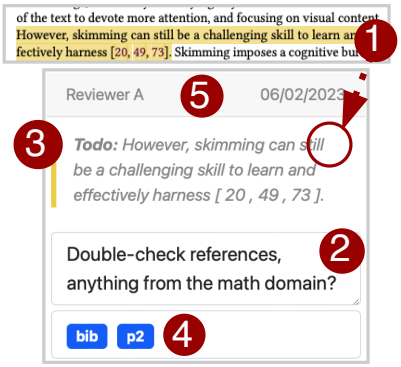

Figure 2: An inline commentary in CARE consists of a highlight (1) optionally associated with a commentary text (2), a label (3), a number of free-form tags (4) and metadata (5), e.g. user name and creation time.

line commentaries on PDF documents in an intuitive manner, connected to a model-agnostic and flexible AI assistance interface. Unlike existing labeling tools, CARE provides a (1) familiar, task-neutral environment for collaborative reading similar to the tools used in everyday text work; unlike off-the-shelf reading and writing applications, CARE offers (2) structured machine-readable data export functionality, including both inline commentary and behavioral data; unlike task-specific AI-assisted reading tools, CARE features a (3) generic interface for integrating NLP modules to support reading and real-time text collaboration.

Our contribution has multiple audiences. For NLP researchers, CARE makes it possible to efficiently collect inline commentary data in a standardized manner, and provides a generic interface for the extrinsic evaluation of NLP models. For application designers, CARE offers an extensible platform and behavioral metrics to study how humans interact with texts. For users, CARE enables the development of new, innovative applications built around AI-assisted reading such as interactive e-learning, community-based fact-checking, and research paper assessment. To foster the progress in AI-assisted reading research, we make the implementation openly available and easy to deploy and to extend.

2 Background

2.1 Terminology and Requirements

The term "annotation" allows for broad interpretation and encompasses both the results of controlled annotation studies that enrich text with a specific new information layer (e.g. named entities), and the less-regulated, natural annotations that humans produce when they work with text. Yet, the two annotation mechanisms are fundamentally different. Labeled NLP data is usually obtained via annotation studies – supervised campaigns that involve formalized tasks and labeling schemata, detailed guidelines, and are supported by specially designed annotation software that requires training of the annotators. However, collecting natural annotation data requires the opposite: the process should minimally interfere with the user's workflow, and the tool should provide a natural environment for working with text given the task at hand. Our work addresses annotation in the latter sense. To avoid ambiguity, we propose the term *inline commentary* to denote in-document highlights left by the users while reading and collaborating on text, potentially associated with commentary text, tags and metadata (Figure 2). We reserve the term *labeling* for the traditional NLP markup.

With this distinction in mind, for a tool to support the study of inline commentary we define the following **requirements** distributed among the key USER GROUPS:

A. Natural environment: The tool should provide the READER with a natural reading environment, specified as allowing the READER to (A1) leave inline commentaries on the documents in (A2) common reading formats like PDF, while requiring (A3) minimal to no training.

B. Collaboration: The tool should run (B1) online and support (B2) real-time collaboration where the READERS can leave, see and reply to each others' commentaries in an on-line fashion.

C. Data management: The tool should enable RESEARCHERS, APPLICATION DEVELOPERS and ADMINISTRATORS to easily (C1) import new documents, (C2) collect inline commentary and USER behavior data, and (C3) export this data in a machine-readable format for further scrutiny.

D. Openness and extensibility: Both documents and inline commentaries might contain confidential data. It is thus crucial that a tool can be (D1) self-hosted on-premise and allows controlling user access to the data. AI-assisted reading has many potential use cases, stressing the need for (D2) high configurability and easy deployment of the tool. To promote transparency and facilitate extension, the platform should be available as (D3) open-source.

E. AI assistance: Finally, the tool should provide an easy way to (E) integrate AI assistance modules

for RESEARCHERS and DEVELOPERS to support USERS in reading and comprehending text.

2.2 Related tools

We identify four broad groups of software tools falling within the scope of our requirements, which we briefly exemplify below. Our overview demonstrates the wide use of inline commentary "in the wild" and underlines the limitations of the existing solutions for the systematic study of inline commentary and AI-assisted reading in NLP.

Readers Highlighting and inline commentary are core features of most standalone reading tools, from PDF viewers like Adobe Acrobat Reader[3] to literature management software like Mendeley[4]. The most commonly used tools are proprietary and thereby hard to extend, and do not offer management, collection and export of fine-grained data, making them unsuitable for the study of inline commentary. While a few dedicated reading applications like ScholarPhi (Head et al., 2021), Scim (Fok et al., 2022), SciSpace[5], and Scrible[6] do offer machine-aided reading assistance, they focus on their particular use cases, lack data collection functionality and extensibility, and can not be easily hosted on-premise to protect potentially sensitive user and document data.

Social annotation Focusing on the collaborative aspect of reading, social annotation platforms allow users to exchange their inline commentaries via a centralized platform. A prime example is Hypothes.is[7], which offers a natural environment, is available open-source and provides a standardized mechanism for exporting inline commentary. Yet, the platform is not easy to extend and customize, and does not offer a standardized mechanism for integrating AI-assistance or behavioral data collection. While not being based on Hypothes.is, CARE adopts many of its design ideas, including the appearance and functionality of the annotation sidebar, utilities to locate inline commentaries in the document text, as well as the underlying data structure of the annotations.

Authoring tools Inline commentary is featured in many text authoring tools, from standalone office applications like Microsoft Office[8] to collaborative web-based platforms like Google Docs[9] and Overleaf[10]. While widely used and familiar, these applications are hard to tailor to the needs of a particular scientific study, offer limited data export capabilities, lack flexible AI integration for assistance, and are either implemented as standalone desktop applications (impeding real-time collaboration), or do not allow self-hosting, making ethical data collection and storage challenging.

Labeling tools The rapid progress in NLP of the past decades has been accompanied by the evolution of general-purpose tools used to acquire labeled data (Neves and Ševa, 2019), from early desktop applications like *WordFreak* (Morton and LaCivita, 2003) to modern extensible, web-based, open-source environments like *brat* (Stenetorp et al., 2012), *labelstudio*[11], *doccano*[12] and *INCEpTION* (Klie et al., 2018). CARE inherits many concepts from NLP annotation platforms – including coupling of external recommenders (Klie et al., 2018), tag sets and study management functionality, and flexible data export. Although not specifically designed for controlled labeling scenarios, CARE can be used as a lightweight labeling tool with collaboration and assistance capabilities.

3 Platform Description

CARE addresses the gap in existing solutions that prevents the study of inline commentary and AI-assisted reading. Here we review the main components of CARE from the user perspective, while the next Section outlines the key technical aspects of our open implementation. We discuss the components in order of importance and refer to the Appendix A for the illustration of a typical user journey.

At the core of CARE is the **reading component** which allows users to attach inline commentaries to documents. To ensure that the visual representation of the document remains true to its source and stable across platforms, CARE focuses on PDF as the main source format[13]. An inline commentary can amount to a simple highlight attached to a continuous text span, can be associated with a free-text

[3] https://www.adobe.com/acrobat/pdf-reader.html
[4] https://www.mendeley.com
[5] https://typeset.io
[6] https://www.scrible.com
[7] https://web.hypothes.is
[8] https://www.office.com/
[9] https://www.google.com/docs/about
[10] https://www.overleaf.com
[11] https://labelstud.io
[12] https://github.com/doccano/doccano
[13] While support for other document formats is planned, we note that any textual document can be converted into a PDF.

note, and can carry a label from a pre-configured label set, as well as any number of free-form tags (Figure 2). It is possible to add document-level commentaries that are not attached to a span. Inline commentaries are displayed in the dedicated **CARE sidebar** and can be navigated and edited. The process is **collaborative**: multiple users can leave inline commentaries on the same document and reply to them in real time. The commentaries are saved and can be revisited at a later point; the resulting data can be **exported** in an easy-to-use data format, individually or in aggregate, and displayed within the user interface of CARE. In addition to the textual data, CARE collects and exports basic behavioral metrics; for instance, the time of highlight creation and the users' scrolling behavior within the document.

The second key component of CARE is **AI assistance**: the inline commentary data can be routed to an arbitrary external NLP module, which returns the prediction that can be displayed in the annotation component *in close-to-real-time* as labels, inline commentary replies, or via a custom UI. The interaction between users and AI assistance is mediated by a flexible **broker** system that distributes the processing tasks among a set of NLP models. Multiple AI assistance model instances can be acting simultaneously, and the pool of models can be extended easily through registering a new model node at the broker backend. At the moment of writing, CARE provides examples to supports integration of any pre-trained model compatible with the *huggingface transformers* API (Wolf et al., 2019) by simply changing the configuration parameters. The model then has access to the inline commentary text, highlighted span from the main document, labels, tags and metadata. It is possible to adapt CARE to use models based on other frameworks.

Finally, CARE features a flexible and configurable **dashboard** that provides quick access to user and system settings, document and label set management, and study management. In particular, the **user management** component is responsible for registration, authentication and authorization; to encourage responsible data collection and ensure that the collected inline commentary data can be used in research, CARE features sample **informed consent** forms that users are presented upon registration, along with the necessary **licensing disclaimers**, which can be refined by the study administrator.

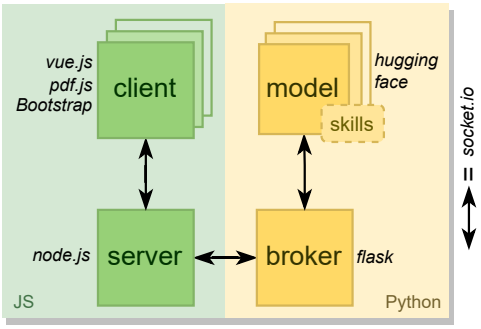

Figure 3: Overview of CARE system architecture.

4 System Design

CARE is designed to be generic, modular and extensible (Figure 3). The ability to build and deploy CARE via a Docker container makes it easy to set it up in new environments. While CARE features detailed documentation, here we provide a high-level overview of the system design. CARE follows a client-server architecture, preferring client-side operation whenever possible to speed up execution and reduce network traffic and server load. This results in a clear separation of responsibilities between the client, the server and the NLP assistance components of CARE and affords high modularity. While the main client-server pair is purely JavaScript-based, the AI models and the broker are implemented in Python to facilitate the NLP assistance development by the natural language processing community.

The web-based **CARE client** is responsible for frontend rendering and annotation functionality. The client is fully implemented in *vue.js*[14], allowing dynamic rendering, modular frontend structure and reuse of original and third-party components. *Bootstrap*[15] is used throughout the frontend to ensure consistent styling and responsive design; document rendering is handled via *pdf.js*[16]. In addition, we adopt the localization code from hypothes.is[17] to locate inline commentaries in the document. The **CARE server**, in turn, is responsible for synchronizing the data among clients, authentication and authorization, and for connecting to external services, including the AI-assistance broker. In line with the JavaScript-based frontend, the backend is implemented in *node.js*[18] as a cross-platform run-

[14] https://vuejs.org
[15] https://getbootstrap.com
[16] https://mozilla.github.io/pdf.js
[17] https://github.com/hypothesis/client
[18] https://nodejs.org/en

time environment. As keeping message transition time low is crucial for collaboration and AI assistance, we base all communication on the *WebSocket protocol*[19]. Persistent bidirectional connection between the client and server components enables real-time exchange of messages and reduces communication time to the possible minimum by reducing the number of connection setups (i.e., three-way handshakes).

AI assistance in CARE is implemented by routing user requests to separately hosted NLP models abstracted into **Skills**: high-level machine-readable specifications of assistance functionalities including inputs, outputs and model configurations (Baumgärtner et al., 2022). The current implementation of NLP assistance in CARE is built on top of the *huggingface* pipeline, making it easy to integrate a wide range of pre-trained models; we provide sample code to facilitate building self-registering docker containers for NLP model deployment. The interactions between the server and the NLP models is mediated by a **Broker** system which distributes user requests among NLP models depending on the necessary skill.

5 User Study

To evaluate and refine the reading environment of CARE in the context of a collaborative applied task (requirements A and B), and to ensure the data export functionality (C) and the extensibility (D) of the system, we have extended the base configuration of CARE to accommodate a custom reading scenario and conducted a user study. We describe the core components of the study here and refer to the Appendix B.1 for details.

Task Scholarly peer review is a prototypical example of close reading accompanied by note-taking, where an expert assesses a manuscript in terms of its originality, readability, validity and impact (Jefferson et al., 2002). We adopted critical reading that takes place during peer review as a basis for our task. The participants of the study were provided with a manuscript-to-review and instructed to leave self-contained annotations on the manuscript while reading. To incentivize reviewers to perform the task rigorously, we simulated a subsequent acceptance-decision-making phase based on the provided annotations. To support the scenario, we extended CARE to allow reviewer-paper assignment and decision-making functionality.

[19] https://datatracker.ietf.org/doc/html/rfc6455

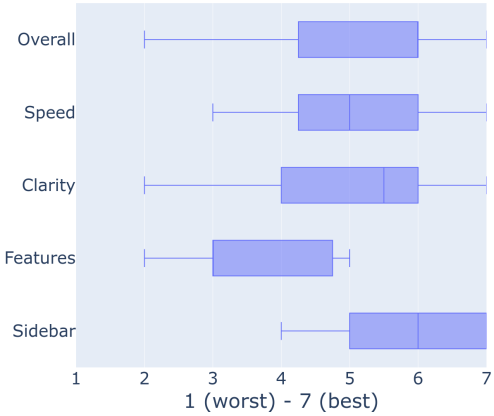

Figure 4: Usability questionnaire results.

Study design We selected two nine-page papers (P1 and P2) from the F1000RD corpus (Kuznetsov et al., 2022), both dedicated to broad academic topics that are understandable for participants with academic background. Before the study, the participants were instructed about the task, and given 15 minutes to familiarize themselves with the CARE environment. The participants were then split into two groups and assigned paper P1 or P2 based on their group. The participants proceeded to review their assigned paper individually under time constraints (40 minutes), following to the task definition provided above. After the time elapsed, the papers were exchanged between the groups, and the participants were asked to make an acceptance decision for the unseen paper given the inline commentaries produced by a reviewer from the other group. The task was performed in English.

Participants In total 11 researchers from the digital humanities (6) and social sciences (5) participated in the study. A pre-study questionnaire verified that the participant demographics were diverse and that more than 60% of the researchers were at a post-doctoral or professorial level in their careers with adequate English proficiency.

Usability After the study, we conducted a usability survey including a subset of the standardized PSSUQ questionnaire (Borsci et al., 2015), as well as free-form questions (details in Appendix B.2). As Figure 4 shows, the majority of participants were satisfied with using CARE for their task and found that the tool provided adequate speed. Most reported that CARE was clear and easy to use, and

appreciated the sidebar functionality. The survey revealed a few feature requests including the ability to arrange inline commentaries in the sidebar by different criteria, and the ability to leave annotations on figure elements.

Data: Inline commentaries The export functionality allowed us to examine the data resulting from the study. In total, participants created 200 inline commentaries of which 151 were associated with commentary text, 17 ± 7.08 commentaries per user per document on average. The highlight spans comprise of on average 161 ± 151.09 characters and vary vastly from single words up to full paragraphs, selections of two to three sentences being the most common. The associated commentaries have 80 ± 109.98 characters on average, ranging from very short remarks of a single word (e.g. "references?", "why?") to full summarizing paragraphs. These results demonstrate the variability of natural inline commentary use.

Data: Reading behavior Behavioral metrics integrated into CARE allowed us to observe how the participants used the tool to perform the task at hand. We observed that 35 annotations (17.5%) were deleted after creation, prompting us to improve the inline commentary edit functionality in the tool; nearly all participants (70%) made use of the ability to quick-scroll from the in-text highlights to the annotations in the sidebar, while the opposite direction (quick-scroll to the highlight *from* the sidebar) was only used rarely. The page tracking functionality allowed insights into how participants assessed the papers: by measuring the time spent on each respective page, we established that the participants spent the least amount of time reading bibliography, whereas method and conclusion sections received most scrutiny. We elaborate on these results in the Appendix B.3.

6 CARE and AI Assistance

Data collection CARE enables the collection of inline commentary data that can be used to study inline commentaries and to create new datasets for NLP assistance model development. The collaboration functionality of CARE allows gathering the data about reader interactions within the tool, and the support for free-form tagging and controlled labeling offers great opportunities for collecting user-generated silver data for model pre-training and fine-tuning.

Assisted reading Out of the box, CARE supports integration of any pre-existing *huggingface transformer* model into the reading workflow, which opens a wide range of possibilities for applying previously developed models "in the wild". To provide feedback to the reader, a pre-trained model can be used to enrich inline commentaries with labels, i.e. prompting the reader to provide additional detail, assessing the politeness (Danescu-Niculescu-Mizil et al., 2013), specificity (Li and Nenkova, 2015) or sentiment (Blitzer et al., 2007) of a commentary. In addition, the power and flexibility of modern generative models like T5 (Raffel et al., 2020) allow performing a wide range of text-to-text tasks to assist reading, from question answering to summarization of highlighted passages, with the results rendered as automatically generated replies to the user's inline commentaries. The CARE repository provides sample code for NLP model integration.

Extrinsic evaluation Finally, the behavioral metrics provided by CARE allow to study both how humans read and comment on documents, and how AI assistance impacts this behavior, for example by recording the order in which parts of the document get accessed, or the time needed to create the commentaries. While the current implementation only supports basic time- and location-based measurements, we envision a wide range of extensions that would help us study the impact of AI assistance on reading and text work.

7 Conclusion and Future Work

This paper has presented CARE – a new open platform for the study of inline commentary and AI-assisted reading. CARE enables efficient inline commentary and behavioral data collection for NLP, and supports a wide range of collaborative reading scenarios, while requiring minimal effort to use. The extensible NLP assistance interface allows using CARE for rapid prototyping and extrinsic evaluation of NLP modules that support reading and text-based collaboration. Planned extensions of CARE include support for non-PDF document processing and automatic text highlighting, improved human-in-the-loop functionality and scalability, as well as further development of the onboard behavioral metrics. We invite the community to use our tool and contribute to its further

development[20].

Ethics

The experiments performed in this study involved human participants who gave explicit consent to the study participation and to the storage, modification and distribution of the collected data. The arbitrary username selection by the users ensured that the behavioral data did not allow any association with the participants unless they decided to reveal this information. We report the demographic distribution of the participants in the Appendix. The documents used in the study are distributed under an open license. Although we have attempted to reflect the reading-for-peer-review workflow as closely as possible, we note that the study might still not be fully representative of the reading practice during peer review, as the participants were strictly limited in time to perform the task, and the selected papers were not necessarily from the participants' domains of specialist expertise.

Any application of AI to assisting humans in performing real-world tasks bears risk. We stress the need to control for bias, harmful content and factuality of the AI models used to assist reading and text work – especially in the case of large pretrained generative models. We deem it equally important to educate the users of AI-assisted reading tools about the limitations and risks associated with the integrated assistance models.

From the data collection perspective, we note that all data collected with CARE is human-generated personal data, in particular the behavioral data. We thus *require* the users of the tool to provide explicit informed consent on the data collection upon registration. In addition, the users must explicitly agree with the optional collection of behavioral statistics before any of this data is transferred to the server (opt-in). We stress that while sufficient for controlled studies, in a real application environment these measures would need to be extended by allowing the users to change their decision at a later point and specify the parts of their data that are included into data collection.

From the privacy perspective, CARE allows registration with an arbitrary username, first and last name, e-mail and password. The choice and management of the usernames and user identities are left to the study administrator – we note that if the usernames are not assigned at random and associated with additional data, this needs to be incorporated into the informed consent form upon registration. CARE implements standard security measures to protect the data, and complete access to the data (documents, inline commentaries, behavioral data) is restricted to the application and server administrator. The security mechanism of the broker and thus of the AI-assistance is currently set via a token defined during the installation of the platform. It is up to the administrator to ensure that the token is kept private, otherwise the models can be used by unwanted users. We stress that for some application scenarios – e.g. dealing with sensitive or confidential documents or performing advanced behavioral measurements – additional security measures should be considered to protect the data.

[20] https://github.com/UKPLab/CARE

Acknowledgements

This work has been funded by the German Research Foundation (DFG) as part of the PEER project (grant GU 798/28-1) and by the German Federal Ministry of Education and Research and the Hessian Ministry of Higher Education, Research, Science and the Arts within their joint support of the National Research Center for Applied Cybersecurity ATHENE.

This research was conducted in the context of a fellowship at the Center for Advanced Internet Studies (CAIS).

Funded by the European Union (ERC, InterText, 101054961). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

References

Tim Baumgärtner, Kexin Wang, Rachneet Sachdeva, Gregor Geigle, Max Eichler, Clifton Poth, Hannah Sterz, Haritz Puerto, Leonardo F. R. Ribeiro, Jonas Pfeiffer, Nils Reimers, Gözde Şahin, and Iryna Gurevych. 2022. UKP-SQUARE: An online platform for question answering research. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 9–22, Dublin, Ireland. Association for Computational Linguistics.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In

Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 440–447, Prague, Czech Republic. Association for Computational Linguistics.

Simone Borsci, Stefano Federici, Silvia Bacci, Michela Gnaldi, and Francesco Bartolucci. 2015. Assessing user satisfaction in the era of user experience: Comparison of the sus, umux, and umux-lite as a function of product experience. *International journal of human-computer interaction*, 31(8):484–495.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Cristian Danescu-Niculescu-Mizil, Moritz Sudhof, Dan Jurafsky, Jure Leskovec, and Christopher Potts. 2013. A computational approach to politeness with application to social factors. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 250–259, Sofia, Bulgaria. Association for Computational Linguistics.

Raymond Fok, Andrew Head, Jonathan Bragg, Kyle Lo, Marti A Hearst, and Daniel S Weld. 2022. Scim: Intelligent faceted highlights for interactive, multi-pass skimming of scientific papers. *arXiv:2205.04561*.

Amir Grinstein and Roy Treister. 2017. The unhappy postdoc: a survey based study. *F1000Research*, 6(1642).

Andrew Head, Kyle Lo, Dongyeop Kang, Raymond Fok, Sam Skjonsberg, Daniel S Weld, and Marti A Hearst. 2021. Augmenting scientific papers with just-in-time, position-sensitive definitions of terms and symbols. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–18.

Tom Jefferson, Elizabeth Wager, and Frank Davidoff. 2002. Measuring the quality of editorial peer review. *Jama*, 287(21):2786–2790.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2022. Survey of hallucination in natural language generation. *ACM Comput. Surv.*

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The INCEpTION platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9, Santa Fe, New Mexico. Association for Computational Linguistics.

Ilia Kuznetsov, Jan Buchmann, Max Eichler, and Iryna Gurevych. 2022. Revise and Resubmit: An Intertextual Model of Text-based Collaboration in Peer Review. *Computational Linguistics*, 48(4):1–38.

Junyi Li and Ani Nenkova. 2015. Fast and accurate prediction of sentence specificity. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1):2281–2287.

Thomas Morton and Jeremy LaCivita. 2003. WordFreak: An open tool for linguistic annotation. In *Companion Volume of the Proceedings of HLT-NAACL 2003 - Demonstrations*, pages 17–18.

Mariana Neves and Jurica Ševa. 2019. An extensive review of tools for manual annotation of documents. *Briefings in Bioinformatics*, 22(1):146–163.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *arXiv:2203.02155*.

Rik Peels, Rene van Woudenberg, Jeroen de Ridder, and Lex Bouter. 2019. Academia's big five: a normative taxonomy for the epistemic responsibilities of universities. *F1000Research*, 8(862).

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Lorenz Stangier, Ji-Ung Lee, Yuxi Wang, Marvin Müller, Nicholas Frick, Joachim Metternich, and Iryna Gurevych. 2022. TexPrax: A messaging application for ethical, real-time data collection and annotation. *arXiv:2208.07846*.

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 *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, Avignon, France. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's transformers: State-of-the-art natural language processing. *arXiv:1910.03771*.

A Application details

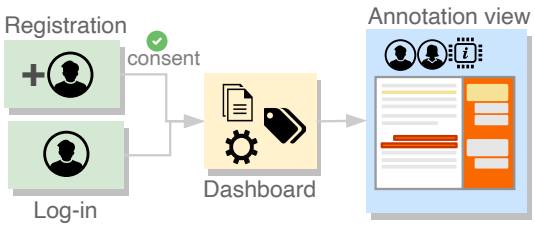

Figure 5: User journey in CARE

User journey Figure 5 illustrates a typical user journey for the reader using CARE. It starts with log-in or registration during which consent and licensing forms are submitted. Afterwards, the user is presented with a dashboard where they can manage documents, label sets and additional settings and export inline commentary and behavioral data. Each document can be opened for reading via the annotation component, where multiple users can annotate the document by leaving inline commentaries organized in a sidebar which also serves as an interface for AI assistance.

Export data format Figure 6 provides an example of the data export functionality: all annotations, comments and discussion threads created by the readers can be directly exported as an easy-to-use JSON. Note that the information presented in the export is also the information available to NLP assistance models to make their predictions.

Behavioral data Figure 7 provides examples of behavioral data that is captured within CARE and exported as JSON objects. Each action is associated with a unique type, meta-data, user information and a timestamp. The captured user interactions include the creation of inline commentary, editing of the same, page scrolling, clicks on important buttons and navigation within the tool.

B User Study Details

This section provides extensive details on the user study setup and results. To recap, the participants were instructed to use CARE for leaving inline commentaries on a manuscript with the purpose of assessing the manuscript's quality and scientific merit, similar to the critical reading process that takes place during scholarly peer review. Participants were split into two groups that reviewed one manuscript each. Participants subsequently exchanged the reviewed manuscripts and used the provided inline annotations to decide whether a manuscript should be accepted or rejected, similar to traditional peer review, and surveyed. Figure 8 summarizes the study design. The papers considered were "Academia's Big Five" (Peels et al., 2019) (P1) and "The Unhappy Postdoc" (Grinstein and Treister, 2017) (P2), both in their first version submitted to F1000 Research.

User Study Context The user study was implemented as a workshop on 25 August 2022 within the Center for Advanced Internet Studies (CAIS)[21]. CAIS is an interdisciplinary research institute in Bochum, Germany, that focuses on the social opportunities and challenges of the digital transformation. Research is conducted in longer-term research programs, as well as by fellows and working groups who are invited to the institute to pursue their own projects. The scientific focus lies on the interface between social sciences, humanities and computer sciences.

B.1 Participant Pool

The participant pool for the user study consisted of 11 CAIS members attending the workshop either virtually (2 participants) or in person (9 participants). No selection criterion was applied to the voluntary participant pool. To ensure the privacy of the participants, we report accumulated frequencies for appropriate value intervals in the following paragraphs.

Demographics Of this participant pool five (45%) identified as women, five (45%) as men and one preferred not to share this information. Around 30% of participants report an age below 40, while the majority of participants lie in the $40-49$ (45%) age range. The rest of the participants (25%) either lie in the age group above 50 or did not report their age. The majority of participants lived and worked in Germany (80%). We deem the given sample as sufficiently diverse for the purpose of this study, as it covers various age groups and shows nearly balanced genders. However, the age group below forty is under-represented, which might have an influence on the study results, as this particular group might show higher digital affinity. Follow-up studies are required to confirm our findings, where a focus on lower age groups and more diverse nationalities should be considered to account for cultural

[21] https://www.cais-research.de

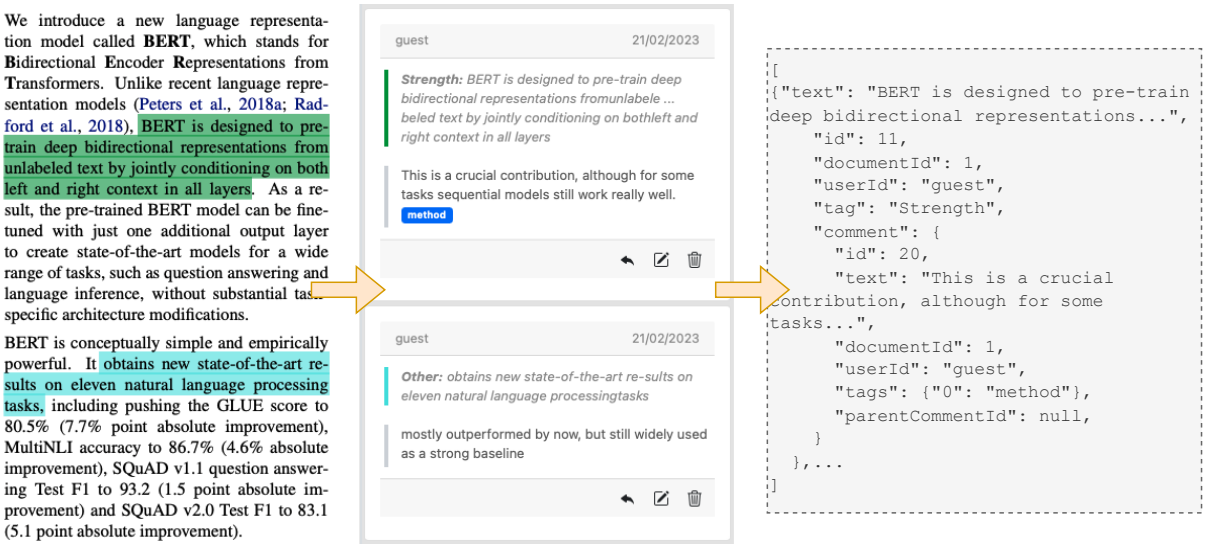

Figure 6: Data export example from highlights to annotations in the sidebar, to export JSON.

differences of the partially subjective peer review assessment process.

Academic Background At the moment of the study, more than 60% of the participants were at a post-doctoral or professorial level in their careers, ensuring an adequate level of expertise and experience in scholarly text work. The participants came from diverse academic backgrounds including social studies, philosophy, law, natural language processing and literary studies. The vast majority of participants (90%) had no computer science background.

Reviewing Expertise In an independent pre-study survey among the CAIS members, we confirmed that English language papers and reviews are the predominant form of scientific communication in their respective fields, suggesting adequate language proficiency of the participants during the study.

Roughly 64% of the participants personally reviewed more than one paper in the past year; only two participants reviewed no papers during their career so-far (zero reviews in the past five years). On average the participants reviewed roughly three papers per year. Apart from the prevalent high academic seniority, these numbers generally suggest deep expertise in the task of peer review, while at the same time the study includes participants with little to no reviewing experience.

B.2 Post-study Survey

The participants were asked to fill out the post-study questionnaire directly after the user study. Each participant responded to the web form individually and privately. We ensured the right of erasure under GDPR regulations[22] and hosted the questionnaire and resulting data exclusively on EU servers. The questionnaire contained in total 35 items structured into the sections demographics and experience and usability.

Quantitative Results The usability section consists of five general usability questions answered on a seven-point scale ranging from "Strongly disagree" (1) to "Strongly agree" (7), as well as free form questions about missing features and feedback about specific design choices. Figure 4 shows the answer distribution on the usability questionnaire. We asked participants to rate the overall experience using CARE, the speed of usage, the ease of finding information, the comprehensiveness of features and the utility of the sidebar.

Qualitative Results Further on, we asked the participants whether they would prefer different orderings of the comments in the sidebar, where the default during the study was an ordering by text position. While this default is perceived as useful (36%), the option for changing the comment order or other grouping strategies are of interest to the users – especially in the decision making phase based on the inline comments of a reviewer. Subsequently, we asked users to highlight which features

[22] https://gdpr-info.eu/art-17-gdpr

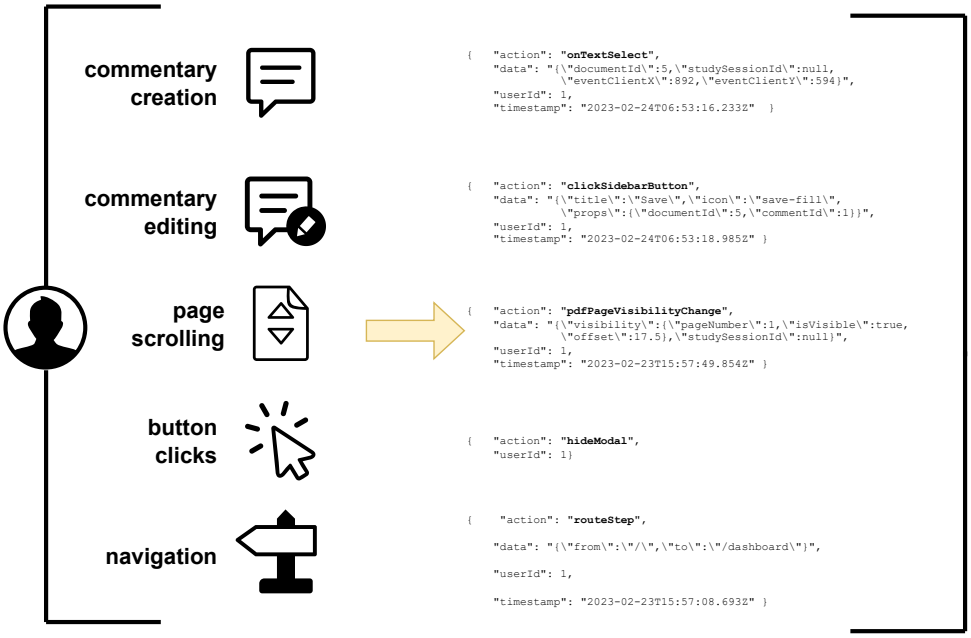

Figure 7: Behavioral user data examples captured and exported as JSON objects.

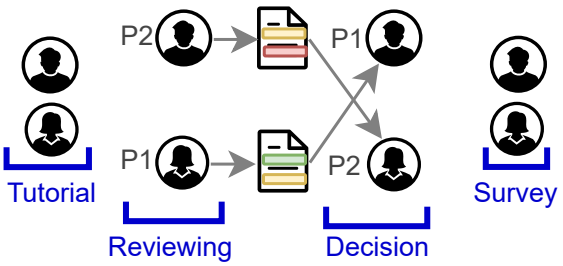

Figure 8: User study setup: Split into two groups after a brief tutorial, the participants review a paper, exchange reviews and make an acceptance decision for the other paper, and participate in the survey.

they missed or could think of to streamline their inline peer review. Most requests were directed towards performing a full peer review based off the inline commentary, e.g. providing notes to editors, providing ratings, having the reviewing guidelines integrated in the interface, etc. Further suggested features that were more focused on the actual highlighting and commenting aspects rather than inline peer review, comprised of more extensive PDF viewer features, like zooming, and improved highlighting features, e.g. sentence-boundary aware highlighting, figure selection, and cross-linking of commentary.

B.3 Behavioral Data

In this section we report on the detailed results of the behavioral data tracking during the user study. Besides showcasing the behavioral data tracking capabilities of CARE, we intend to collect insights into usage patterns of the tool, as well as establishing a deeper understanding of the use-case of assisting reviewers during reading.

Task Timing We consider several timing metrics to measure the ease of usage, as well as the task difficulty.

First, we measure the time-to-completion, starting with the users accessing the document and ending with them submitting their inline review. The median time-to-completion amounts to 37.82min (just below the provided time limit), with a high standard deviation of roughly 13min. Except for two outliers requiring below 15min, this suggests most people did use and require the full time interval to perform their inline peer review.

Second, we measure the time passed before the interaction with a feature of CARE was registered. This includes text selections for highlights, scrolling to a new page, or creating a comment in the sidebar. We employ this metric as an indicator for the bandwidth of the perceived user interface complexity. In fact, we see that on median 1.28min pass before the first interaction, while again showing high standard deviation of 50s. The high variance and relatively long median time before the first interaction suggest that some participants were still familiarising with the study instructions while already having accessed the document. This shows

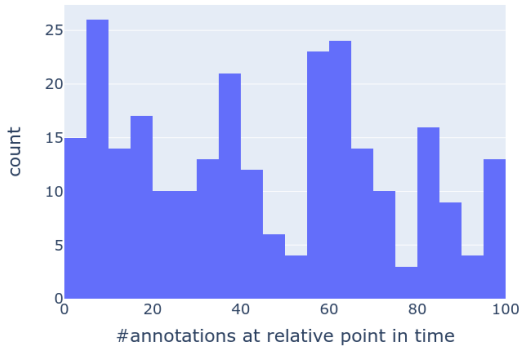

Figure 9: Histogram of the distribution of annotations across time relative to the user's task timing. We accumulate across users.

one limitation of the behavioral tracking implemented in CARE so-far: while the behavioral data logging is non-obstructive to the user experience, unlike e.g. eye-tracking devices in laboratory scenarios, off-screen activities such as breaks cannot be detected reliably.

Reading and Inline Commentary We consider two metrics to analyze the participants' focus of attention during the reading process. As the first metric, we consider the time of inline commentary creation relative to the total task time, to quantify whether participants create annotations throughout the reading process or detached before or after reading. For an inline commentary x created at t_c we define the *reltime* relative to the user's time of entering the document t_e and the time of leaving the document t_l as:

$$\textit{reltime}(x) = \frac{t_c(x) - t_e}{t_l - t_e}$$

Figure 9 shows the distribution of relative inline commentary timings across participants. Apparently the participants create annotations throughout the whole annotation process, with a light dip at 50%, i.e. after half of the time to completion. These measurements do not suggest that making inline commentary is decoupled from the actual reading process, instead CARE seems to support regular highlighting and note-taking habits while reading.

Turning to the second metric, we compute the time elapsed while viewing a page during the study. We compute the relative reading time per participant and page, considering the two papers in isolation. To estimate the relative time spent per page, we measure the time deltas between two subsequent page view events, indicating that a PDF page has been rendered on the participants screen, and normalize by the total task time. While this metric is a sufficient approximation for the purpose of assessing the overall reading coverage throughout the document, the measurements on page level instead of scrolling positions limit fine-grained claims about the reading position of a user.

Figure 10 shows the median reading times per page of the users for the two papers in isolation. For both papers, the reading times have a similar "M" shape, where the least amount of time is spent on the very first page, the middle part of the paper and the final pages. For P2 we observe a consistent peak on page two containing the main part of the introduction and, with high variance, page six including the discussion and a central figure of the article. For P1 individual page reading times are less pronounced, but we see peaks on page three (including a large table) and the pages five and six consisting of a long body of text explaining the core contribution (a taxonomy) of the paper.

In the given user study setting, the page viewing times may reveal the parts of the paper that received most scrutiny during reading and commenting, as well as an estimate of the coverage of all paper aspects by the participants. For instance, we see that the bibliography has not been analyzed in detail by any of the participants. In general scenarios, the page viewing times may reveal places of interest in a document or indicate passage that require more effort to process during reading.

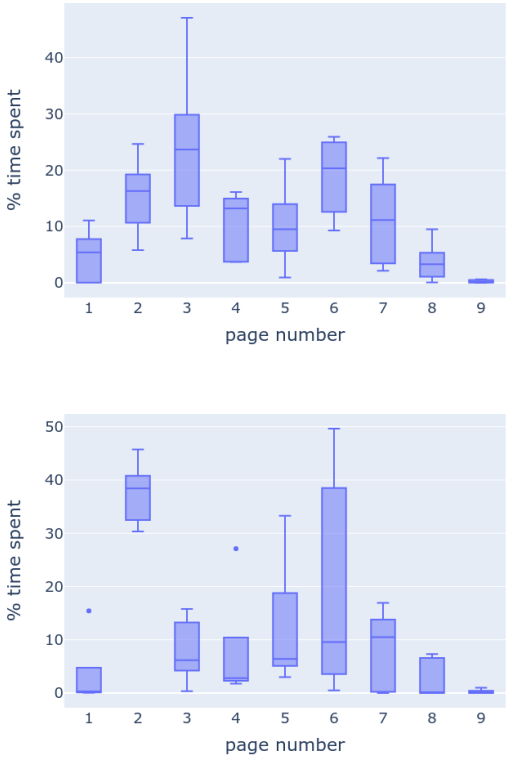

Figure 10: Relative reading time per page for papers P1 (top) and P2 (bottom)