INLG 2025

Proceedings
of the
18th International Natural Language Generation Conference

System Demonstrations

October 29 – November 2, 2025

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
209 N. Eighth Street
Stroudsburg, PA 18360
USA
Tel: +1-570-476-8006
Fax: +1-570-476-0860
acl@aclweb.org

ISBN 979-8-89176-322-7

This event is sponsored by Vingroup Innovation Foundation (VINIF – VinBigData).

Organizing Committee

Program Chairs

Lucie Flek, University of Bonn / Lamarr Institute of Machine Learning and Artificial Intelligence
Shashi Narayan, Google DeepMind
Lê Hồng Phương, VNU Hanoi University of Science
Jiahuan Pei, Vrije Universiteit Amsterdam

Generation Challenge Chair

Simon Mille, ADAPT Research Centre / Dublin City University

Workshop & Tutorial Chair

Xiaoyong Wei, The Hong Kong Polytechnic University

Local Organizer Chairs

Nguyễn Lê Minh, Japan Advanced Institute of Science and Technology
Nguyễn Thị Minh Huyền, VNU Hanoi University of Science

Local Organizers

Nguyễn Việt Cường, Intelligent Integration Co., Ltd. (INT²)
Đỗ Văn Hải, Thuy Loi University
Phan Xuân Hiếu, VNU University of Engineering and Technology
Lê Hoàng Quỳnh, VNU University of Engineering and Technology
Nguyễn Phương Thái, VNU University of Engineering and Technology
Nguyễn Minh Tiến, Hung Yen University of Technology and Education
Nguyễn Thị Thu Trang, Hanoi University of Science and Technology
Trần Đức Vũ, Japan Advanced Institute of Science and Technology

Publication Chair

Ondřej Dušek, Charles University

Sponsor Chair

Shreyas Sharma, aiXplain

SIGGEN Executives

Chenghua Lin, University of Manchester
David M. Howcroft, University of Aberdeen
Saad Mahamood, Shopware
Simon Mille, ADAPT Research Centre / Dublin City University
Patrícia Schmidtová, Charles University

Area Chairs

Reinald Kim Amplayo, Google
Guanyi Chen, Central China Normal University
Ondřej Dušek, Charles University
Sebastian Gehrmann, Bloomberg LP
Kelvin Han, Independent
Rudali Huidrom, ADAPT Research Centre / Dublin City University
Johannes Kiesel, Bauhaus-Universität Weimar
Lara J. Martin, University of Maryland, Baltimore County
Laura Perez-Beltrachini, University of Edinburgh
Ehud Reiter, University of Aberdeen
Fahime Same, trivago N.V.
João Sedoc, New York University
Sina Zarrieß, University of Bielefeld

Program Committee

Rim Abrougui, Aday
Alyssa Allen, The Ohio State University
Miriam Anschütz, Technical University of Munich
Mary-Jane Antia, University of Cape Town
Xin Bai, Central China Normal University
Anya Belz, ADAPT Research Centre / Dublin City University
Nadjet Bouayad-Agha, NLP Scientist
Daniel Braun, University of Marburg
Gordon Briggs, U.S. Naval Research Laboratory
Alberto Bugarín-Diz, University of Santiago de Compostela
Eduardo Calò, Utrecht University
Thiago Castro Ferreira, Federal University of Minas Gerais
Guanyi Chen, Central China Normal University
Chung-Chi Chen, National Institute of Advanced Industrial Science and Technology
Wei-Fan Chen, University of Bonn
Watson Wei Khong Chua, Government Technology Agency of Singapore
Miruna Adriana Clinciu, Edinburgh Centre for Robotics
Hugo Contant, Carnegie Mellon University
Brian Davis, Dublin City University
Simon Dobnik, University of Gothenburg
Shen Dong, University of Glasgow
Pablo Duboue, Textualization Software
Ondřej Dušek, Charles University
Lucie Flek, University of Bonn / Lamarr Institute of Machine Learning and Artificial Intelligence
Albert Gatt, Utrecht University
Ray Groshan, University of Maryland, Baltimore County
Kelvin Han, Independent
Kei Harada, The University of Electro-Communications
Hiroaki Hayashi, Salesforce Research
Arya Honraopatil, University Of Maryland Baltimore County
David M. Howcroft, University of Aberdeen
Nikolai Ilinykh, University of Gothenburg
Simeon Junker, Bielefeld University
Mihir Kale, Meta

Srinivas Ramesh Kamath, trivago N.V.
Yoshinobu Kano, Shizuoka University
Debanjana Kar, IIT Kharagpur
Zdeněk Kasner, Charles University
Gary Kazantsev, Bloomberg
Emiel Krahmer, Tilburg University
Lea Krause, Vrije Universiteit Amsterdam
Mateusz Lango, Poznan University of Technology / Charles University
Guy Lapalme, RALI-DIRO Université de Montréal
Ashley Lewis, The Ohio State University
Jin Li, Shenzhen Institute of Advanced Technology / Chinese Academy of Sciences
Stephan Linzbach, GESIS - Leibniz Institute for the Social Sciences
Wei-Yun Ma, Academia Sinica
Saad Mahamood, Shopware
Zola Mahlaza, University of Cape Town
Aleksandre Maskharashvili, University of Illinois Urbana-Champaign
Kathleen F. McCoy, University of Delaware
David D. McDonald, Smart Information Flow Technologies
Gonzalo Méndez, Universidad Complutense de Madrid
Qingyu Meng, Vrije University Amsterdam
Antonio Valerio Miceli Barone, The University of Edinburgh
Simon Mille, ADAPT Research Centre / Dublin City University
Yifan Mo, Vrije Universiteit Amsterdam
Anna Nikiforovskaya, CNRS/LORIA / Université de Lorraine
Kristýna Onderková, Charles University
Suraj Pandey, The Open University
Siyana Pavlova, Université de Lorraine
Jiahuan Pei, Vrije Universiteit Amsterdam
Pablo N Perez De Angelis, Gezie.io
Minh Vu Pham, IT:U Austria
Toky Hajatiana Raboanary, University of Cape Town
Vandana Sreenivasa Rao, Microsoft
Ehud Reiter, University of Aberdeen
Philipp Sadler, University of Potsdam
Fahime Same, trivago N.V.
Daniel Sanchez, University of Granada
Sashank Santhanam, University of North Carolina at Charlotte / Apple
Patrícia Schmidtová, Charles University
Anastasia Shimorina, Orange
Judith Sieker, Bielefeld University
Adarsa Sivaprasad, University of Aberdeen
Yifei Song, CNRS-LORIA
Yingjin Song, Utrecht University
William Eduardo Soto Martinez, LORIA
Somayajulu Sripada, Arria NLG / University of Aberdeen
Prerak Srivastava, SAP Labs
Symon Stevens-Guille, The Ohio State University
Kristina Striegnitz, Union College
Yue Su, Vrije Universiteit Amsterdam
Barkavi Sundararajan, University of Aberdeen
Jan Svec, Brno University of Technology

Hiroya Takamura, The National Institute of Advanced Industrial Science and Technology
Ekaterina Taktasheva, University of Edinburgh
Marc Tanti, University of Malta
Mariet Theune, University of Twente
Ilias Triantafyllopoulos, New York University
Qingyun Wang, William & Mary
Robert Weißgraeber, AX Semantics
Hugh Mee Wong, Utrecht University
Siwei Wu, Nanjing University of Science & Technology
Xinnuo Xu, Microsoft Research
Bohao Yang, University of Manchester
Kun Zhang, INRIA Saclay / École Polytechnique
Huajian Zhang, Westlake University
Tianyi Zhang, University of Pennsylvania
Ingrid Zukerman, Monash University
Rodrigo de Oliveira, IQVIA
Chris van der Lee, Tilburg University

Best Area Chairs

Ondřej Dušek
Sebastian Gehrmann
Kelvin Han
Laura Perez-Beltrachini
Ehud Reiter
Fahime Same

Best Reviewers

Miriam Anschütz
Nadjet Bouayad-Agha
Chung-Chi Chen
Kathleen F. McCoy
Kelvin Han
David M. Howcroft
Lea Krause
Ashley Lewis
Wei-Yun Ma
Toky Hajatiana Raboanary
Fahime Same
Qingyun Wang

Table of Contents

Echoes of Others: Real-Time LLM Dialogue Generation for Immersive NPC Interaction

James McGrath and Michela Lorandi
Dublin City University
{james.mcgrath, michela.lorandi}@mail.dcu.ie

Anya Belz
Dublin City University
anya.belz@dcu.ie

Abstract

Large Language Models (LLMs) promise unscripted, adaptive NPC dialogue, but their latency and resource demands hinder real-time deployment in games. Our aim is to demonstrate how viable it is, to have low-latency NPC conversations that run on consumer hardware and to characterise the speed–quality trade-offs between local and cloud models. We introduce Echoes of Others, an Unreal Engine 5 prototype that integrates three back-ends—(i) GPT-4o Mini (cloud), (ii) OpenHermes-7B, and (iii) a LoRA-tuned 4-bit variant trained on 100k lines of RPG dialogue—via a lightweight server. The system runs on consumer hardware while maintaining a 60 FPS budget and dynamic response generation. We evaluate latency and dialogue quality across three RPG scenarios using LLM-as-a-Judge scoring on fluency, relevance, and persona consistency.

1 Introduction and Background

Modern role-playing games rely on scripted dialogue, limiting player choice and replayability despite massive writing efforts. Baldur's Gate 3, for example, contains over 125,000 hand-authored lines, yet players are still constrained to fixed options, making conversations predictable. Unscripted, generative Non Player Characters (NPC) dialogue can preserve character and world consistency while enabling unanticipated questions, creating more immersive experiences without the heavy authoring costs.

Recent advances in LLMs have opened new possibilities for dynamic, unscripted dialogues. While traditional systems in titles like *Mass Effect* or *Skyrim* rely on finite-state machines or branching scripts, research prototypes such as *Façade* (Mateas and Stern, 2003) and *NPCEditor* (Leuski and Traum, 2010) have explored procedural and statistical approaches. However, the complexity and resource demands of such systems limited their practical adoption.

Figure 1: Screenshot of the in-game town with a dialogue interaction.

Integrating LLMs into modern engines like Unreal Engine 5 (UE5) introduces new technical challenges, such as latency, memory usage, and maintaining dialogue coherence in real-time. Performance constraints need careful optimisation, including level-of-detail scaling, occlusion culling (Epic Games, 2023), and the Nanite geometry engine (AMD GPUOpen, 2022), to free GPU capacity for inference tasks.

To support character consistency, prompt engineering plays a central role. Conditioning LLMs with persona descriptions, world lore, and stylistic constraints, drawing on work like PersonaChat (Zhang et al., 2018) and Generative Agents (Park et al., 2023), helps sustain in-character responses across dialogue turns.

In this paper, we propose a UE5 working prototype that enables real-time dialogue generation for NPCs. A backend bridge connects to local or cloud-based back-ends, generating character-aware responses on consumer hardware. We evaluate trade-offs between model quality, latency, and system responsiveness, and offer a replicable blueprint for developers. A video demo is available at https://youtu.be/uvoi5wA7rpc.

2 System Components

Gameplay. The game follows classic role-playing design: players are free to explore, complete quests, and influence the world state through their actions. With the introduction of LLM-based dialogue, ev-

Proceedings of the 18th International Natural Language Generation Conference: System Demonstrations, pages 1–2
October 29–November 2, 2025.

ery character has its own distinct personality and equal possibilities for unique interactions without the need for thousands of handcrafted lines. As players progress, changes in the world state (e.g., completed quests or character deaths) dynamically alter persona prompts, creating new opportunities for context-aware interactions.

Interactive Dialogue Flow. Dialogue generation is triggered by a UE5 Blueprint node that collects (a) the player's utterance, which is checked against a list of banned words and terms before being passed to the LLM to prevent toxic content, (b) the chat history, and (c) the character's personality and general information about the game world, which is dynamically updated based on the current world state. These elements are used to construct the prompt given to the LLM, which is bundled into a JSON payload and transmitted via the `HttpRequest` subsystem. The server returns a structured response; only the main reply is shown in the game UI. The system supports a single active speaker at a time; concurrency will be implemented in future work.

System Architecture. Figure 2 illustrates the overall system architecture for dialogue interactions. A lightweight inference server connects UE5 to local or cloud LLM back-ends, supporting hot-swapping without restarting the game. Local models are loaded with `BitsAndBytesConfig` using 4-bit NF4 quantisation and merged LoRA adapters. Safety is enforced via a regex-based filter. The entire pipeline is designed for drop-in backend replacement and minimal performance overhead.

Model Fine-Tuning and Persona Adaptation. LoRA (Hu et al., 2021) is used to fine-tune the base LLM, and the trained module is merged into the frozen model at inference time. The chat history is truncated client-side to manage token limits, and persistent world state (e.g., quest flags) is used to adjust persona prompts dynamically.

To fine-tune the pretrained LLM, we used transcribed scripts of Skyrim and The Witcher, two game of the year winning open world RPG games. After cleaning and chunking into overlapping 1024-token windows, we generated 10.7M prompt–completion pairs. We further annotated lines with high-level roles (e.g., *Guard*, *Merchant*, *Farmer*) and subsampled 10,000 examples per role to ensure persona diversity. The supervised objective is standard causal language modelling so that the model learns to generate the next in-character turn conditioned on recent dialogue and persona text.

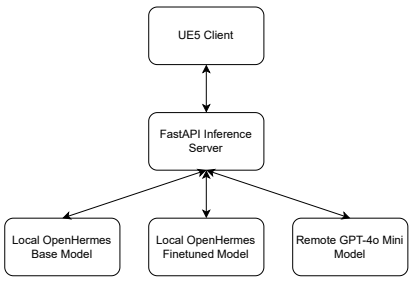

Figure 2: Real-time inference pipeline.

Evaluation. Across three live RPG contexts (*Guard*, *Blacksmith*, *Priest*) we compared GPT-4o Mini, OpenHermes-7B, and a LoRA-tuned OpenHermes. Each model was scored by three independent LLM-as-a-judge on relevance, persona consistency, and fluency, with latency measured separately. Mean server latencies were 1.9 s, 12.3 s, and 3.0 s respectively. Command-R Plus judge mean scores (1–10) were 8.7, 7.0, and 4.7.

References

AMD GPUOpen. 2022. Nanite and geometry optimization in UE5. https://gpuopen.com/learn/unreal-engine-performance-guide/.

Epic Games. 2023. Visibility and occlusion culling. https://dev.epicgames.com/documentation/en-us/unreal-engine/visibility-and-occlusion-culling-in-unreal-engine.

Edward J. Hu and 1 others. 2021. LoRA: low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Anton Leuski and David Traum. 2010. NPCEditor: a tool for building question-answering characters. In *Proc. of the 7th Intl. Conf. on Intelligent Virtual Agents (IVA)*.

Michael Mateas and Andrew Stern. 2003. Façade: an experiment in building a fully-realized interactive drama. In *Game Developers Conference (GDC)*.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: interactive simulacra of human behavior. In *Proceedings of the 36th ACM Symposium on User Interface Software and Technology (UIST)*.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: "I have a dog, do you have pets too?". In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

CSPaper Review: Fast, Rubric-Faithful Conference Feedback

Lele Cao
King (Part of Microsoft)
lelecao@microsoft.com

Lei You
Technical University of Denmark
leiyo@dtu.dk

R&D Team
CSPaper Review*
rnd@cspaper.org

Abstract

CSPaper Review (CSPR) is a free, AI-powered tool for rapid, conference-specific peer review in Computer Science (CS). Addressing the bottlenecks of slow, inconsistent, and generic feedback in existing solutions, CSPR leverages Large Language Models (LLMs) agents and tailored workflows to deliver realistic and actionable reviews within one minute. In merely four weeks, it served more than 7,000 unique users from 80 countries and processed over 15,000 reviews, highlighting a strong demand from the CS community. We present our architecture, design choices, benchmarks, user analytics and future road maps.

1 Why We Built It

Two pressing challenges have emerged in the fast-growing landscape of Computer Science (CS) research conferences, especially in AI and Machine Learning (ML). First, novice researchers often lack timely, targeted feedback tailored to their chosen conferences, with useful input arriving too late (typically after rejection) to guide meaningful revision. Second, the surge in submissions to top venues like ICML and NeurIPS has overwhelmed the traditional peer review system, leading to delays, inconsistent assessments, and declining review quality (Kim et al., 2025; Guo et al., 2023; Naddaf, 2025). As a result, reviewer capacity is stretched thin, compromising the depth and consistency of evaluations.

While Large Language Models (LLMs) are already quietly assisting with peer reviews – Liang et al. 2024a estimate that 6.5%∼16.9% of reviews at top AI conferences were ghostwritten or substantially revised by GPT-4 or alikes – the existing AI review tools fails to address the needs of paper authors representing a broader CS community.

CS stands out from other scientific disciplines in *three key ways* that makes it particularly suitable for AI-assisted reviewing. First, CS has evolved into a vast and fast-moving field where **conference publications dominate over journals** due to their strict timelines and rapid dissemination cycles. CS researchers therefore have a much stronger demand for **early feedback** to improve and iterate quickly. Second, CS conferences typically publish **well-defined and standardized review rubrics**, offering a natural scaffolding for aligning LLM-generated reviews with human expectations. This structured evaluation format is rare in other disciplines, making CS an ideal testbed for AI feedback. Third, the CS community is highly **active, decentralized, and open**, with an unmatched culture of preprints, open-source projects, and community-driven innovation. Fourth, some top-tier AI conference officially starts introducing AI-assisted review as a supplement to human reviewers (AAAI, 2026). This **strong communal foundation** is essential for "Human+AI" review systems.

However, existing tools such as Rigorous, WBS, GroundedAI, PaperWizard, and Hum fail to meet these CS-specific needs: they target journal workflows (not conference-style reviewing), take days to respond, lack rubric-aligned ratings, are prohibitively expensive and often tuned for non-CS domains like biology. To address this gap, we introduce CSPaper Review (CSPR), a **free** (up to 20 reviews per day) LLM-powered paper review system built from the ground up for CS researchers, with conference-specific evaluation criteria, fast turnaround, and integration into a researcher's early feedback loop.

2 How It Works

CSPR accepts either arXiv IDs/URLs or directly uploaded PDFs. Within **60 seconds**, the platform generates conference-specific reviews comprising three sections: *desk rejection assessment*, *expected review outcome*, and *critical reviewer ratings*.

Latex/PDF processor: As depicted in Fig 1, dedicated processors extract text, tables, equations, and images from both LaTeX source packages and PDF files. For LaTeX inputs, the system performs downloading, main-tex resolution, consolidation of scattered tex files, and content cleaning. PDF inputs undergo OCR parsing, generating structured JSON content composed of markdown and images.

*CSPaper Review: https://review.cspaper.org. The R&D Team also includes Kai Xie, Weiping Ding, Yong Du, Sven Salmonsson, Yumin Zhou, and Vilhelm von Ehrenheim.

Proceedings of the 18th International Natural Language Generation Conference: System Demonstrations, pages 3–7
October 29–November 2, 2025.

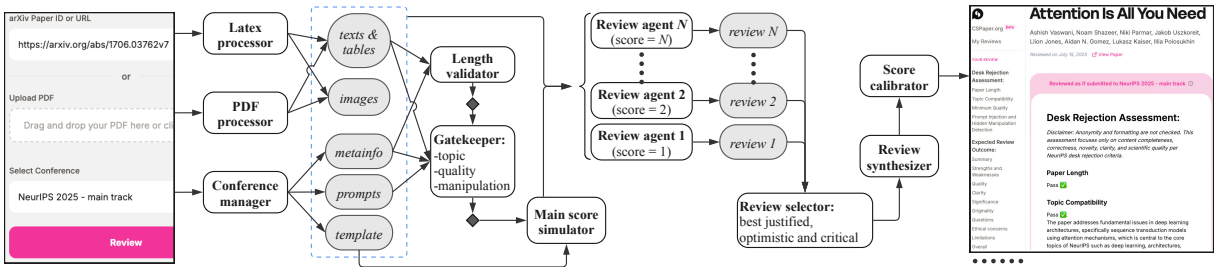

Figure 1: Input, output and workflow of CSPaper Review (CSPR). White boxes represent components or agents; gray boxes represent intermediate artifacts. Small diamonds indicate points where a failure in the preceding step terminates the entire workflow.

Images are normalized to JPEG and downsampled (when excessively large) to ensure sensible LLM token consumption while keeping visual clarity.

Conference/track manager stores and retrieves conference-specific meta-information (name, track, year, call for papers, deadlines, etc.), review templates, and curated review prompts (with examples) tailored to individual conferences and tracks. It ensures generated reviews adhere strictly to the standards and expectations of selected venues.

Pre-review checks: Extracted artifacts are sequentially evaluated through a paper length validator and a set of gatekeepers verifying topic relevance, overall quality, and risk of prompt manipulation. Any failure at this stage immediately terminates the review process.

Review agents: For each valid rating/score level defined by the target conference (e.g., *1-strong reject* to *5-strong accept*), we **force** a dedicated agent to (concurrently) generate reviews that strictly justify the assigned score/rating. A review selector identifies three most realistic reviews: best justified, more optimistic, and more critical. They are synthesized into a coherent output primarily based on the best-justified review but selectively incorporating insights from the other two versions. Finally, a calibration step ensures coherence between overall and sub-dimensional scores (e.g., novelty, clarity), ensuring a well-aligned and balanced final review.

3 What We Found

LLM choice: We constructed a benchmark dataset of 100 papers by manually collecting reviews from OpenReview, official conference websites, and social media. We evaluated five LLMs on this benchmark. Mean Absolute Error (MAE) is calculated using the ground-truth overall scores as labels. The model with the lowest MAE (cf. Table 1 in Appendix) was selected as the serving LLM.

PDF parser: We qualitatively compared 4 PDF parsers (MinerU, Rigorous, Mistral, and LandingAI) on five CS papers with varied layouts. Mistral stood out with clean, structured JSON and highly accurate transcription of text, tables, equations, algorithms and images, while MinerU and Rigorous produced frequent, review-impacting errors. LandingAI showed similar quality to Mistral but is less viable due to pricing and speed.

Step-by-step vs. all-in-one prompting is a key question in LLM research; while step-by-step approaches are thought to enhance reasoning (Yu, 2024), explicit decomposition can sometimes harm performance (Liu et al., 2024b). In our experiments, splitting each review agent into specialized sub-agents did not improve MAE, but increased token usage fivefold and latency over tenfold.

User analytics: Most traffic came from referral (44%), followed by direct (36%) and organic search (28%). Referral users were the most engaged, with a 3-minute average session and 48% of total page views. The number of arXiv and PDF review requests is largely equal. Among the 162 users who participated in our survey, 64% were undergraduate, graduate, or postgraduate students, consistent with the findings of (Liang et al., 2024b). We identified three notable usage patterns: 1) the same paper reviewed across multiple conferences/tracks, likely to determine the most suitable submission venue; 2) different versions of a paper reviewed within the same conference/track, suggesting iterative improvement of writing; and 3) one-time PDF review requests where filenames include real conference submission IDs, potentially indicating use of the tool for self-assessment during the review process. Please refer to the Appendix for additional results and ethical discussions.

4 What's Next

CSPR has demonstrated real-world value in streamlining CS paper reviews, and our goal is to evolve it into both a practical tool and a research testbed for advancing human-AI collaboration in peer review. We aim to broaden coverage, enhance agent capabilities, develop interactive interfaces, and imple-

ment safeguards for trustworthy AI-assisted reviewing. Ultimately, we seek to benefit CS researchers while advancing the theory and practice of computational research assessment.

References

AAAI. 2026. AAAI-26 Main Technical Track: Call for Papers. Accessed: 2025-08-25.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The Llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

Yanzhu Guo, Guokan Shang, Virgile Rennard, Michalis Vazirgiannis, and Chloé Clavel. 2023. Automatic analysis of substantiation in scientific peer reviews. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10198–10216. Association for Computational Linguistics.

Jaeho Kim, Yunseok Lee, and Seulki Lee. 2025. Position: The AI conference peer review crisis demands author feedback and reviewer rewards. In *Forty-second International Conference on Machine Learning Position Paper Track*.

Weixin Liang, Zachary Izzo, Yaohui Zhang, Haley Lepp, Hancheng Cao, Xuandong Zhao, Lingjiao Chen, Haotian Ye, Sheng Liu, Zhi Huang, and 1 others. 2024a. Monitoring ai-modified content at scale: A case study on the impact of chatgpt on ai conference peer reviews. In *International Conference on Machine Learning*, pages 29575–29620. PMLR.

Weixin Liang, Yuhui Zhang, Hancheng Cao, Binglu Wang, Daisy Yi Ding, Xinyu Yang, Kailas Vodrahalli, Siyu He, Daniel Scott Smith, Yian Yin, and 1 others. 2024b. Can large language models provide useful feedback on research papers? a large-scale empirical analysis. *NEJM AI*, 1(8):AIoa2400196.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Ryan Liu, Jiayi Geng, Addison J Wu, Ilia Sucholutsky, Tania Lombrozo, and Thomas L Griffiths. 2024b. Mind your step (by step): Chain-of-thought can reduce performance on tasks where thinking makes humans worse. In *Forty-second International Conference on Machine Learning*.

Miryam Naddaf. 2025. AI is transforming peer review — and many scientists are worried. *Nature*, 639(8056):852–854.

Yijiong Yu. 2024. Do LLMs really think step-by-step in implicit reasoning? *arXiv preprint arXiv:2411.15862*.

Appendix of
CSPaper Review: Fast, Rubric-Faithful Conference Feedback

A More Figures and Tables

Figure 2, adopted from Google Analytics Dashboards, illustrates the distribution of unique users by country.

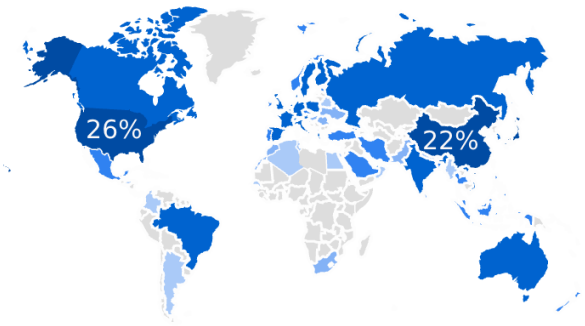

Figure 2: The geographical distribution of over 7,000 unique CSPR users from 80 countries.

Table 1 presents the mean absolute error (MAE) for five LLMs, GPT-4.1, GPT-o3, GPT-o4-mini, Deepseek-v3 (Liu et al., 2024a), and Llama3-8b (Dubey et al., 2024), evaluated across eight conferences, using a benchmarking dataset of 100 carefully selected research papers. We applied the following practices while constructing the dataset:

- For accepted papers, we did not randomly sample from all accepted works. Instead, we prioritized well-received papers such as spotlights, award-winning papers, or those that drove significant community discussion (e.g., on OpenReview, Alphaxiv and social media). These papers are generally considered exemplars in their respective venues and thus represent strong, trusted evaluation anchors.
- We acknowledge that rejected papers are generally harder to obtain, as reviews are often not made public. To address this, we relied on manual sourcing where possible, including data from conferences with open review processes (ICLR and partially NeurIPS) and from authors who are willing to share their rejected work and reviews. This ensured our negative examples came from verifiable, credible sources rather than arbitrary low-quality drafts.
- We explicitly identified cases where the final decision diverged from the average score or where reviewer opinions were highly polarized. In such cases, we asked established senior researchers (not involved in our team) to calibrate the scores, providing a more stable and reliable label for benchmarking. This step directly mitigates the concern that our benchmark might inherit inconsistencies from the review pool.
- Our benchmark dataset was deliberately balanced across multiple top-tier CS conferences and tracks to avoid bias toward a single venue's reviewing style or quality distribution. This diversity helps ensure the evaluation is not overfitted to one conference's reviewing idiosyncrasies.

Conference	GPT-4.1	o3	o4-mini	DS3	Llama3	GPT-5
AAAI	**0.044**	0.077	0.113	0.110	0.170	0.086
CVPR	**0.033**	0.100	0.100	0.082	0.150	0.067
EMNLP	**0.100**	0.160	0.180	0.170	0.210	0.120
ICLR	**0.120**	0.200	0.240	0.230	0.280	0.200
ICML	**0.092**	0.175	0.275	0.263	0.320	0.175
IJCAI	**0.100**	0.125	0.125	0.220	0.280	0.125
KDD	0.188	**0.125**	0.333	0.310	0.390	0.188
NeurIPS	**0.098**	0.131	0.348	0.333	0.395	0.131

Table 1: Benchmarking results to choose serving LLMs. "DS" denotes DeepSeek. Best results are highlighted in **bold**, and second-best results are underlined.

Figure 3 visualizes the distribution of user profiles among the 162 respondents to our questionnaire.

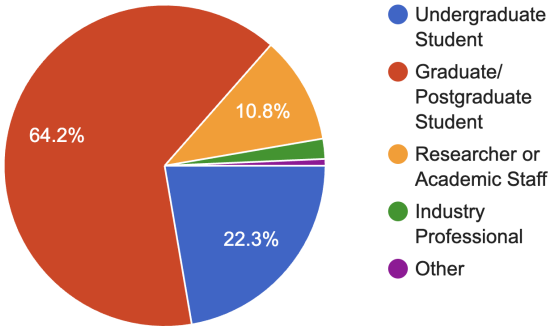

Figure 3: Percentage of user profiles from questionnaire.

Figure 4 presents the frequency of review activities reported by users in their daily work, as

captured by the same questionnaire.

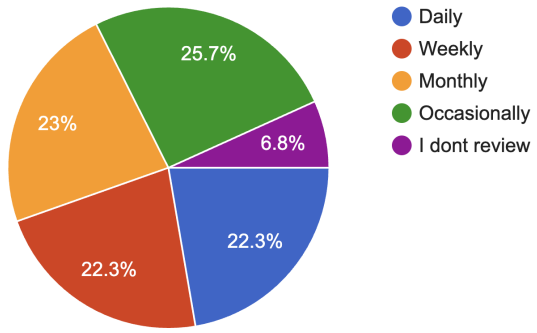

Figure 4: User review frequency from questionnaire.

B Data Handling and Ethical Considerations

CSPaper Review adheres to responsible data handling and transparency principles. All data collected and analyzed in this study (including uploaded PDF manuscript files, selected target conferences, and review preferences) were processed in accordance with our publicly available privacy policy.[1] Manuscript files are processed automatically using LLMs, under contractual agreements that explicitly prohibit the use of submitted content for model training or fine-tuning.

Uploaded files are temporarily stored on secure cloud infrastructure (e.g., Microsoft Azure) and are deleted either upon user request or after a defined expiration period. No user-submitted content is sold, shared, or publicly disclosed. On rare occasions, individual manuscripts may be reviewed internally for debugging and improvement, strictly under secure, privacy-preserving conditions.

User analytics presented in this paper (e.g., referral sources, usage patterns) are aggregated and fully anonymized. No personally identifiable information (PII)[2] is collected or disclosed. Observations such as filenames containing conference submission IDs (e.g., NeurIPS) were recorded passively and are not linked to individual users.

All users provide explicit consent to these practices when submitting their manuscripts. Only minimal cookies are used in a strict way.

[1] https://cspaper.org/assets/uploads/review/privacy-policy.pdf

[2] https://en.wikipedia.org/wiki/Personal_data

C Acknowledgments

We are grateful to William Stoddart, Mathias Holst, and Sylvia Li for their support in the organizational, financial and operational matters.

We thank Orhan Uyaver, Wen Zhou, Filip Jasson and Rui Zhou for their early contributions in verifying the minimum viable product (MVP). We also appreciate Xiaolong Liu (Intel), Wenbing Huang (and his Lab in Renmin University), Yongfeng Zhang (and his Lab in Rutgers University), Heng Fang (KTH), Ye He (UCL), Sofiane Ennadir (KTH and Microsoft), Zineb Senane (Télécom Paris), Fangkai Yang (Microsoft), Valentin Buchner (University of Amsterdam), Tianze Wang (KTH and Microsoft) and Johannes F. Lutzeyer (Ecole Polytechique) for their valuable early evaluations in their respective research domains.

We are also grateful to Alexandra Stark and Tim Elgar from King (part of Microsoft) for reviewing this paper from communication and legal perspectives, respectively.

We thank the INLG 2025 conference reviewers for their constructive feedback, three detailed double-blind reviews and one meta-review, which significantly contributed to the improvement of this manuscript.

Finally, we sincerely acknowledge the feedback and encouragement from the broader CSPaper community, whose engagement has been invaluable to the development of this work.

VitaEval: Open-source Human Evaluation Tool for Video-to-Text and Video-to-Audio Systems

Goran Topić[1] **Yuki Saito**[2] **Katsuhito Sudoh**[3] **Shinnosuke Takamichi**[4]
Hiroya Takamura[1] **Graham Neubig**[5] **Tatsuya Ishigaki**[1]
[1]AIST [2] The University of Tokyo [3] Nara Women's University
[4] Keio University [5] Carnegie Mellon University
{goran.topic, ishigaki.tatsuya}@aist.go.jp

Abstract

We present VitaEval, an open-source tool that streamlines the preparation process for human evaluation of video-to-text and video-to-audio systems. Evaluating such systems typically requires segmenting long videos, aligning subtitles and synthesized audio, and building custom interfaces for annotators—tasks that are time-consuming and often technically demanding. VitaEval addresses these challenges by automating video segmentation, synchronizing audio and subtitles, and generating web-based interfaces. Researchers can deploy evaluation setups without needing expertise in FFmpeg or HTML5, significantly lowering the barrier to conducting multimodal human evaluations. In a case study, we demonstrate that annotation setups using VitaEval can be created within one hour. The demo video is available at https://www.youtube.com/watch?v=TL4w1vWWaNY.

1 Introduction

Video-to-text generation, including dense captioning and real-time commentary systems (Krishna et al., 2017; Ishigaki et al., 2021, 2023), is gaining attention as large language models become increasingly capable of processing and describing dynamic visual content. Recent advances even extend these systems with text-to-speech technologies to produce natural-sounding commentary, enabling immersive experiences for users (Ishigaki et al., 2023). However, conducting human evaluation for such multimodal outputs remains a bottleneck. Evaluators must assess not just text quality, but also the alignment between video, text, and audio. This requires researchers to perform video segmentation, subtitle formatting, audio overlay, and interface design—steps that may seem lightweight individually but accumulate into a significant amount of manual effort and technical expertise in video and web technologies.

Existing annotation tools such as CVAT (Corporation, 2024), VIA (Dutta and Zisserman, 2019), and Label Studio (Tkachenko et al., 2020-2022) are built primarily for vision tasks and lack support for multimodal synchronization or interface automation. Commercial platforms offer flexibility but require extensive customization and cost. Thus, there is a strong need for a lightweight, customizable, and open-source solution tailored to multimodal evaluation.

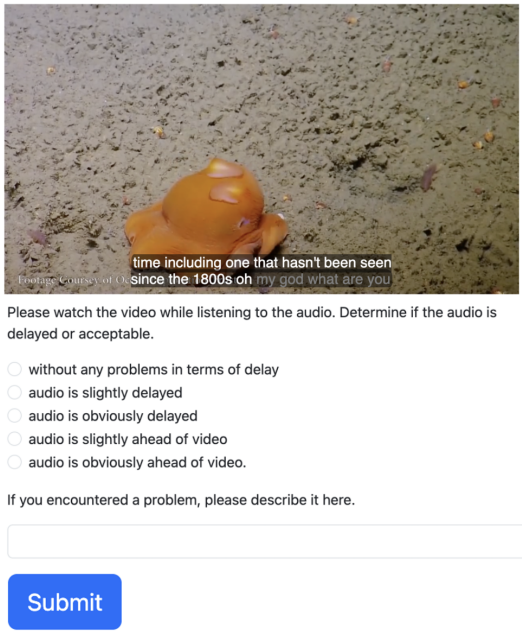

Figure 1: The evaluation interface, automatically generated by VitaEval.

Despite the importance of evaluation in generation research, many existing benchmarks rely on simplified setups where only the text output is evaluated in isolation. However, in realistic applications such as sports commentary, instructional videos, or accessibility services, the interplay between video, text, and audio is crucial for understanding user experience. Manual preparation of evaluation setups for such tasks not only slows down research cycles but also hinders reproducibility and scalability. There is a pressing need for

Proceedings of the 18th International Natural Language Generation Conference: System Demonstrations, pages 8–9
October 29–November 2, 2025.

tools that can lower this barrier and enable rapid, reproducible, and scalable evaluation for multimodal generation systems. Thus, we propose VitaEval.

2 VitaEval: System Overview

VitaEval is a Django-based web application that provides an end-to-end pipeline for setting up human evaluation tasks involving video, text, and audio. It supports two user roles: administrators who configure evaluation projects, and evaluators who annotate the video segments.

In the administrator workflow, to create an evaluation project, administrators upload a video file along with an optional JSON file specifying cut points, subtitle files (SRT/WebVTT), and commentary audio (MP3/WAV). VitaEval uses FFmpeg to cut the video and synchronize overlays. It then generates an evaluation interface where subtitles are rendered via HTML5 captioning and audio commentary is played back in sync with video.

Figure 1 shows an example interface automatically generated by VitaEval. Each evaluation screen presents a video segment with overlaid captions and audio, followed by one or more questions (e.g., "Is the audio delayed?") and optional free-text comments. Question types include radio buttons, checkboxes, and text fields. Results are saved in JSON and downloadable through the admin panel.

For scalability, VitaEval supports integration with Amazon Mechanical Turk (MTurk) for crowdsourcing, AWS S3 for media hosting, and scriptable endpoints for bulk data upload. Multiple administrators can manage tasks collaboratively. The system runs on Python 3.10 and can be hosted locally or on a cloud server.

3 Case Study

We applied VitaEval to annotate a racing game commentary dataset (Ishigaki et al., 2023), aiming to classify utterances as either subjective or objective. We spent 10 minutes to write a python script to convert SRT files to cut intervals, we deployed the evaluation interface within an hour by using VitaEval. In total, our tool reduces manual coding effort by at least 141 lines. The interface allowed annotators to make judgments using synchronized audio and subtitles, showcasing the tool's utility in real-world research workflows.

4 Conclusion

By automating video segmentation, audio/text synchronization, and interface creation, VitaEval eliminates the need for manual scripting or web development. Our system is easy to deploy and supports large-scale evaluation via crowdsourcing platforms and cloud storage. The source code and documentation are publicly available at: `https://github.com/aistairc/commentary_evaluator`. Further technical details can be found in the supplementary material (Topić et al., 2025) [1].

References

CVAT.ai Corporation. 2024. Computer vision annotation tool (cvat).

Abhishek Dutta and Andrew Zisserman. 2019. The via annotation software for images, audio and video. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, page 2276–2279, New York, NY, USA. Association for Computing Machinery.

Tatsuya Ishigaki, Goran Topić, Yumi Hamazono, Ichiro Kobayashi, Yusuke Miyao, and Hiroya Takamura. 2023. Audio commentary system for real-time racing game play. In *Proceedings of the 16th International Natural Language Generation Conference: System Demonstrations*, pages 9–10, Prague, Czechia. Association for Computational Linguistics.

Tatsuya Ishigaki, Goran Topić, Yumi Hamazono, Hiroshi Noji, Ichiro Kobayashi, Yusuke Miyao, and Hiroya Takamura. 2021. Generating racing game commentary from vision, language, and structured data. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 103–113, Aberdeen, Scotland, UK. Association for Computational Linguistics.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, pages 706–715.

Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. 2020-2022. Label Studio: Data labeling software. Open source software available from https://github.com/heartexlabs/label-studio.

Goran Topić, Yuki Saito, Katsuhito Sudoh, Shinnosuke Takamichi, Hiroya Takamura, Graham Neubig, and Tatsuya Ishigaki. 2025. Detailed technical descriptions of VitaEval: A open-source human evaluation tool for video-to-text and video-to-audio systems. In *arXiv*.

[1]This study is based on results obtained from a project in BRIDGE implemented by the Japanese government.

ARTIST: A Learning Support System for Fostering Students' Argumentative Writing Skills

Thomas Huber
University of St. Gallen, Switzerland
thomas.huber@unisg.ch

Christina Niklaus
University of St. Gallen, Switzerland
christina.niklaus@unisg.ch

Abstract

We present ARTIST, a learning support system that can help students assess their argumentative writing and provide automated, individual feedback, thus improving their writing performance. It analyzes student-written argumentative texts by identifying argument components and their relationships. The resulting argumentative discourse structure is displayed in an interactive interface. In that way, the ARTIST tool provides immediate and personalized visual feedback on the quality of students' texts, supporting self-monitoring and reflection on how to improve their texts.

1 Introduction

Argumentative writing skills are essential to enable one to convey one's own understanding and critical thinking. Effort and training are needed to improve them. In many contexts however, writing skills are not promoted and training measures are not very effective (Thaiss and Zawacki, 2006; Stevenson and Phakiti, 2014). However lecturers often do not have time to provide individual feedback to each student. Generic responses hinder their learning progress. This is problematic, as argumentative writing is rarely done outside of schools when one is a student. To address this issue, recent advances in Natural Language Processing (NLP) and Artificial Intelligence (AI) are leveraged to analyze the writing quality of texts and to provide students with personalized and adaptive feedback, as well as to support gains in student's writing motivation and quality (Zhang, 2013; Rapp and Kauf, 2018; Strobl et al., 2019).

Whereas automated support for revisions on the *micro-level*, targeting factual knowledge (e.g., grammar, spelling, word frequencies) is well-represented in current literature, tools that support the development of writing strategies and encourage self-monitoring to improve *macro-level* text quality (e.g. argumentative structures, rhetorical moves) are still rare. Therefore, we propose an AI-enabled learning support system to assess students' argumentative writing and to automate feedback to individual students, thus supporting writing performance. This enables personalized learning. One of the most significant benefits of using AI in education is seen as a support tool for personalized learning and formative feedback (Stone et al., 2016; Zawacki-Richter et al., 2019). ARTIST contributes to this new emerging interdisciplinary research field as recent advances in AI emphasize the importance of better understanding of the human-machine power relationship in learning and problem solving (Wesche and Sonderegger, 2019; Raisamo et al., 2019; Seufert et al., 2020).

We make a video demonstration of ARTIST available at https://youtu.be/f0s2EcWd7fU and release the code at https://github.com/unisg-ics-dsnlp/artist-inlg2025.

2 Interface

ARTIST provides direct and indirect feedback through three main channels: (i) the Argumentation Dashboard, (ii) the Discourse Structure overview, which provides an analysis of the rhetorical structure and coherence relations of the text to help the user identify weaknesses, and lastly (iii) through direct, adaptive Improvement Suggestions. Figure 1 shows the Argumentation Dashboard.

Argumentation Dashboard Claims, major claims and premises are highlighted in different colours directly in the input and presented as a graph, showing the argumentative structure of the text. A detailed view shows how the individual components of each argument connect with each other. A sunburst diagram shows the proportion of how much of the text consists of argumentative components. Coherence and Persuasion scores are presented as a box plot based on the rating

Proceedings of the 18th International Natural Language Generation Conference: System Demonstrations, pages 10–13
October 29–November 2, 2025.

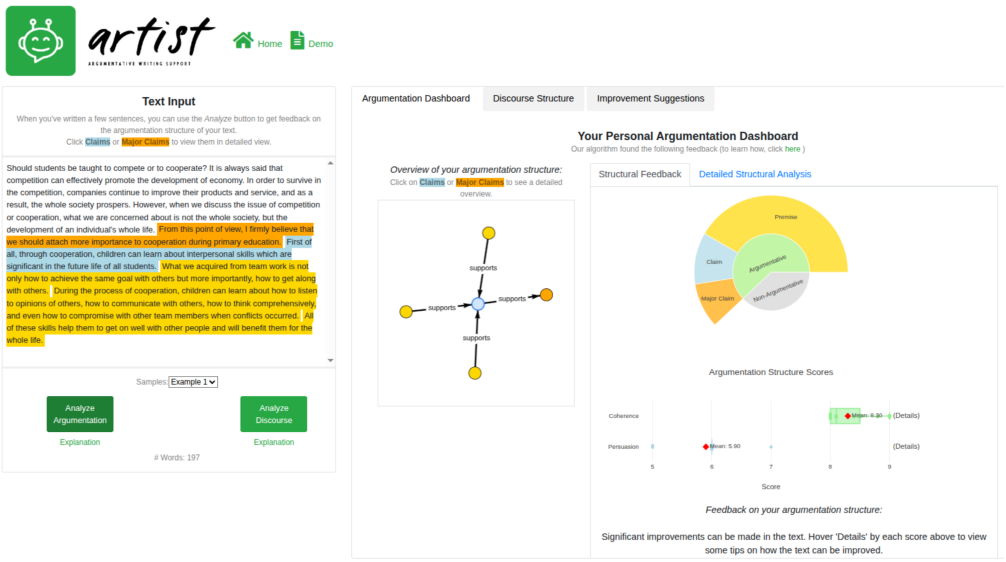

Figure 1: Screenshot of the ARTIST writing support system. A highlighted argument is shown, as well as the structure graph, the sunburst diagram of the distribution of components and the Coherence and Persuasion scores.

of multiple LLM raters following the approach introduced by Hu et al. (2024). To simulate a panel of experts the model is prompted 50 times with a temperature of 0.7. Scores are presented as a boxplot to provide feedback about the consistency of the scores to the user. We use plotly.js for this plot.

Discourse Structure The Rhetorical Structure Theory (RST) (Mann and Thompson, 1988) parse tree of the input text is shown, with explanations for the more high-level discourse markers. The RST tree shows the text split into discourse units and how they relate to each other. By default a simplified version using the high level markers by Fraser (1996) is shown. The complex markers are mapped to *Elaborative*, *Inferential*, *Contrastive* and *Temporal*. Experienced users can switch to 'Expert Mode' to see fine-grained labels. Users can select discourse units to highlight the relation in the text. Explanations for the labels are shown next to the graph. Figure 2 shows the Discourse Structure functionality.

Improvement Suggestions The user can request adaptive improvement suggestions for their text. These suggestions are made by an LLM, and adapt to the user's input.

3 Implementation Details

Backend The backend of ARTIST is a Python Django project.

LLM ARTIST supports using self-hosted LLMs. We use a Llama 3.3 70B instance running on 8 V100 GPUs. We want to emphasize that smaller models, with lower hardware requirements, are suitable alternatives. This includes small models like the Phi family of models, which are designed to be hardware efficient (Abdin et al., 2024a,b), and can be run locally on current consumer grade laptops. The LLM is used for the improvement suggestions feature, as well as to calculate the Coherence and Persuasion scores.

Visualization We use vis.js for the visualization of the argument structure, Plotly.js for the structural feedback, Cytoscape.js for the RST tree.

Discourse Structure Detection We use an updated version of RST parser by Feng and Hirst (2014b,a). The parser itself is unchanged, but we provide a Python package to make it easier to use. The updated package is available at `https://github.com/ThHuberResearch/feng-hirst-rst-parser`.

4 Evaluation

We evaluated successive prototypes of our argumentation feedback system through a series of controlled laboratory experiments and real-world classroom studies, demonstrating its effectiveness in improving students' argumentative skills. For instance, in a study with first-year students (n=80), we observed measurable gains in argumentation competency (Burkhard et al., 2023). In a comple-

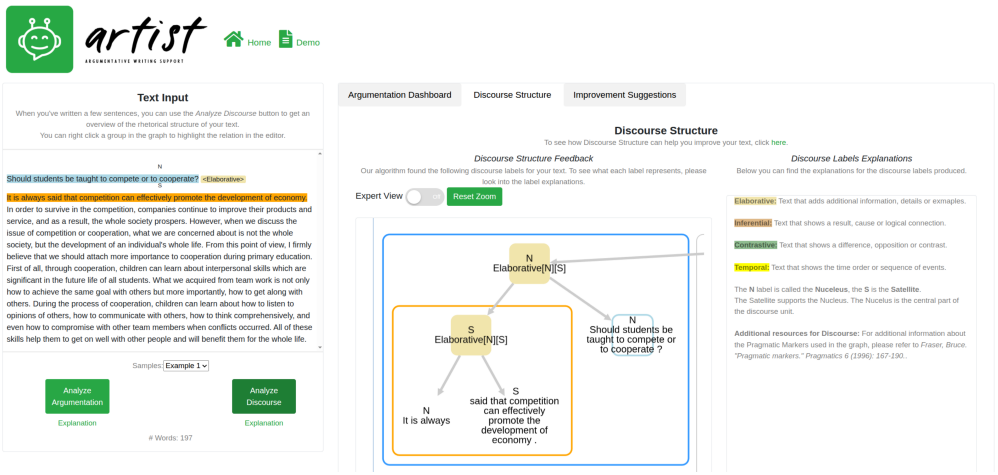

Figure 2: Screenshot of the Discourse Structure functionality of ARTIST. A subgraph of the discourse tree is highlighted, and the relation is shown in the argument on the left. The labels are explained to the right of the graph.

mentary study with 30 participants, we collected qualitative feedback on the tool's usability and perceived effectiveness (Htaw et al., 2024). Moreover, students (n=63) rated the quality of the feedback provided by open-source and proprietary LLMs positively. More precisely, they regarded the suggestions for improving their argumentative texts as helpful (7.51 vs. 7.65 on a 10-point Likert scale) (Gubelmann et al., 2024). Most importantly, students wrote more convincing essays with higher formal argument quality, producing on average 5.1 arguments with our tool compared to 3.2 with a baseline scripting tool (Wambsganss et al., 2020).

5 Example User Interaction

In the following, we describe a typical use case scenario of the ARTIST tool.

The user enters an argumentative text. They click *Analyze Argumentation*. This highlights their argument components, and shows their relationships in the dashboard, which helps find unsubstantiated claims in the text. The *Structural Feedback* sunburst diagram shows that a large portion of the text is non-argumentative. The Coherence and Persuasion scores are also rather low. Next, the user presses *Analyze Discourse*. This generates an RST parse tree, which shows the individual discourse units and their relation. The user is experienced, so they toggle *Expert View*, which provides fine-grained labels. They right-click a subgraph with an *Explanation* and it is highlighted in the text. The user realizes that a part of their text, intended to explain a certain point they were making, does not have this relation in the graph. They read the corresponding passage and note they did not properly elaborate their point. They revise the sentence and generate the graph again. The user analyzes the new graph and is satisfied. Lastly, they open the *Improvement Suggestions* tab, and request individual feedback. The feedback suggests to add concrete examples or evidence to further strengthen the argument. Based on the analyses provided by the tool the user improves their argument further. As the user keeps working with the tool, their overall argumentation skills improve.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, and 110 others. 2024a. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, and 1 others. 2024b. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

Michael Burkhard, Sabine Seufert, Reto Gubelmann, Christina Niklaus, and Patcharin Panjaburee. 2023. Computer supported argumentation learning: Design of a learning scenario in academic writing by means of a conjecture map. In *CSEDU (1)*, pages 103–114.

Vanessa Wei Feng and Graeme Hirst. 2014a. A linear-time bottom-up discourse parser with constraints and post-editing. In *Proceedings of the 52nd Annual*

Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 511–521, Baltimore, Maryland. Association for Computational Linguistics.

Vanessa Wei Feng and Graeme Hirst. 2014b. Two-pass discourse segmentation with pairing and global features. *Preprint*, arXiv:1407.8215.

Bruce Fraser. 1996. Pragmatic markers. *Pragmatics*, 6:167–190.

Reto Gubelmann, Michael Burkhard, Rositsa V. Ivanova, Christina Niklaus, Bernhard Bermeitinger, and Siegfried Handschuh. 2024. Exploring the usefulness of open and proprietary llms in argumentative writing support. In *Artificial Intelligence in Education. Posters and Late Breaking Results, Workshops and Tutorials, Industry and Innovation Tracks, Practitioners, Doctoral Consortium and Blue Sky*, pages 175–182, Cham. Springer Nature Switzerland.

Mi Chan Htaw, Daria Pipa, Namkang Sriwattanarothai, Chailerd Pichitpornchai, Reto Gubelmann, Sabine Seufert, Christina Niklaus, and Siegfried Handschuh. 2024. Argumentative writing software: Perceptions of undergraduate students toward artist prototype. In *2024 IEEE 7th Eurasian Conference on Educational Innovation (ECEI)*, pages 92–96.

Zhe Hu, Hou Pong Chan, and Yu Yin. 2024. AMERICANO: Argument generation with discourse-driven decomposition and agent interaction. Association for Computational Linguistics.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Roope Raisamo, Ismo Rakkolainen, Päivi Majaranta, Katri Salminen, Jussi Rantala, and Ahmed Farooq. 2019. Human augmentation: Past, present and future. *International Journal of Human-Computer Studies*, 131:131–143.

Christian Rapp and Peter Kauf. 2018. Scaling academic writing instruction: Evaluation of a scaffolding tool (thesis writer). *International Journal of Artificial Intelligence in Education*, 28(4):590–615.

Sabine Seufert, Josef Guggemos, and Stefan Sonderegger. 2020. Digitale transformation der hochschullehre: Augmentationsstrategien für den einsatz von data analytics und künstlicher intelligenz. *Zeitschrift für Hochschulentwicklung*, 15(1):81–101.

Marie Stevenson and Aek Phakiti. 2014. The effects of computer-generated feedback on the quality of writing. *Assessing Writing*, 19:51–65. Feedback in Writing: Issues and Challenges.

Peter Stone, Rodney Brooks, Erik Brynjolfsson, Ryan Calo, Oren Etzioni, Greg Hager, Julia Hirschberg, Shivaram Kalyanakrishnan, Ece Kamar, Sarit Kraus, and 1 others. 2016. Artificial intelligence and life in 2030. *One Hundred Year Study on Artificial Intelligence: Report of the 2015-2016 Study Panel*, 52.

Carola Strobl, Emilie Ailhaud, Kalliopi Benetos, Ann Devitt, Otto Kruse, Antje Proske, and Christian Rapp. 2019. Digital support for academic writing: A review of technologies and pedagogies. *Computers & Education*, 131:33–48.

C.J. Thaiss and T.M. Zawacki. 2006. *Engaged Writers and Dynamic Disciplines: Research on the Academic Writing Life*. Boynton/Cook.

Thiemo Wambsganss, Christina Niklaus, Matthias Cetto, Matthias Söllner, Siegfried Handschuh, and Jan Marco Leimeister. 2020. Al: An adaptive learning support system for argumentation skills. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–14, New York, NY, USA. Association for Computing Machinery.

Jenny S. Wesche and Andreas Sonderegger. 2019. When computers take the lead: The automation of leadership. *Computers in Human Behavior*, 101:197–209.

Olaf Zawacki-Richter, Victoria I Marín, Melissa Bond, and Franziska Gouverneur. 2019. Systematic review of research on artificial intelligence applications in higher education–where are the educators? *International Journal of Educational Technology in Higher Education*, 16(1):1–27.

Mo Zhang. 2013. Contrasting automated and human scoring of essays. *R & D Connections*, 21(2):1–11.

Author Index