Interaction-Required Suggestions for Control, Ownership, and Awareness in Human-AI Co-Writing

Kenneth C. Arnold
Calvin University
kcarnold@alum.mit.edu

Jiho Kim
Calvin University
jihokim8@acm.org

Abstract

This paper explores interaction designs for generative AI interfaces that necessitate human involvement throughout the generation process. We argue that such interfaces can promote cognitive engagement, agency, and thoughtful decision-making. Through a case study in text revision, we present and analyze two interaction techniques: (1) using a predictive-text interaction to type the assistant's response to a revision request, and (2) highlighting potential edit opportunities in a document. Our implementations demonstrate how these approaches reveal the landscape of writing possibilities and enable fine-grained control. We discuss implications for human-AI writing partnerships and future interaction design directions.

1 Introduction

Current chatbot interfaces for large language models like ChatGPT, Claude, and Gemini limit interaction to a turn-taking conversation, even though the underlying models could support more versatile interactions, especially for writing tasks.

In this paper, we begin to explore the design space of interactions that people can have with model outputs, focusing on the potential opportunities presented by interactions where human initiative is *required* for completing a task. Although these interactions are, by construction, less efficient at producing plausible outputs, we aim to explore the potential benefits they might offer in control, ownership, visibility of the solution space, and feedback for model tuning.

We present two interaction techniques for revision in writing: predictive-text and opportunity highlighting. The first technique adapts the familiar predictive-text interaction (top-k suggestions or free typing) found on mobile devices to allow people to type the *assistant's response* word by word. The second technique visualizes alternative

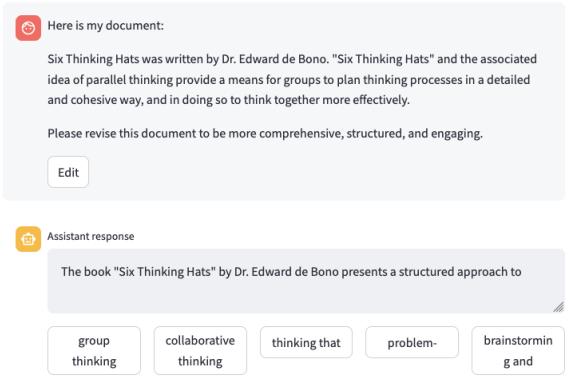

Figure 1: Predictive text interaction repurposed to type the assistant's response

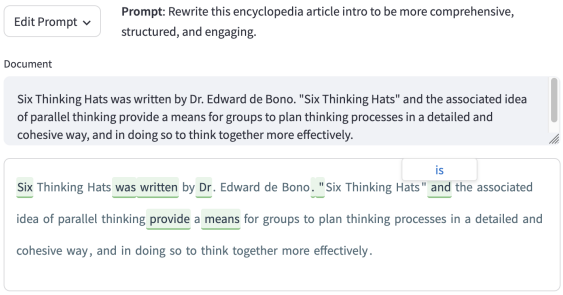

Figure 2: Highlighting opportunities for divergent choices

(and sometimes divergent) choices for revising text according to a writer-specified goal.

2 Design Principles

AI support for writing has evolved primarily along two interaction paradigms: conversational exchanges with an assistant (as in modern chatbots) and editorial feedback systems (like inline markup in Grammarly or reflection tools like Impressona (Benharrak et al., 2024) and Textfocals (Kim et al., 2024). While these paradigms have proven useful, they both place the AI in a position of either generating content or evaluating it, with humans primarily reacting to AI output.

Proceedings of the Fourth Workshop on Intelligent and Interactive Writing Assistants (In2Writing 2025), pages 62–68

We propose interaction-required approaches that fundamentally shift this dynamic by necessitating ongoing human involvement throughout the generation process. Our approach is guided by three design principles that emphasize cognitive partnership between humans and AI systems:

Prioritize cognitive engagement over efficiency Although AI assistance can speed task completion, using it without cognitive engagement can lead to overconfidence (Fernandes et al., 2025), errors (Dakhel et al., 2023), and skill stagnation (Gajos and Mamykina, 2022). Interactions can instead be designed to encourage writers' thoughtful participation rather than optimizing solely for speed or ease. This principle addresses how AI systems can support authentic self-expression, ownership, and accountability in writing, which many writers desire (Biermann et al., 2022; Hwang et al., 2024). The literature on explainable AI systems for decision-making suggests cognitive engagement as valuable goal (Datta and Dickerson, 2023).

Enable granular control Rather than offering only coarse accept/reject options for completed AI outputs, interfaces could instead allow writers to influence the progress of generation. Granularity could enable just-in-time feedback that shapes the direction of AI assistance, providing a way for users to clarify their goals without having to engage in prompt refinement or writing examples.

Reveal the landscape of possibilities Interactions should make visible the alternatives available at each decision point, helping writers understand the range of options and make more informed choices. Prior work has explored contextual suggestions of alternative words or phrases at targeted points (e.g., Reza et al. (2023); Gero and Chilton (2019)), but some authors have explored interfaces for navigating through the tree of suggestions in a narrative generation context (Reynolds and McDonell, 2021).

2.1 Interaction-Required Suggestions

The degree to which a writing support interface *requires* interaction can be measured, in principle, by an *amplification ratio*, the ratio of the entropy of system output (new text or edits) to the entropy of user input. For example, asking a chatbot to write a complete essay or make overall edits has a high amplification ratio since the input entropy is confined to the prompt. Accepting grammar suggestions also has high amplification ratio, since the user often only needs to click "Accept".

We conjecture that LLM-powered interfaces with a low amplification ratio can be designed according to these design principles to assist writers at various points in the writing process.

3 A Case Study in Revision

We will present two interaction designs that embody these design principles for the purpose of revision. As a running hypothetical example, suppose Alex is a Wikipedia editor who wants to revise the introduction section for the article on "Six Thinking Hats", as it was on 2025-02-25:

> "Six Thinking Hats was written by Dr. Edward de Bono. "Six Thinking Hats" and the associated idea of parallel thinking provide a means for groups to plan thinking processes in a detailed and cohesive way, and in doing so to think together more effectively."

We will use a revision instruction generated by Claude.ai: "Rewrite this document to be more comprehensive, structured, and engaging."

3.1 Typing the Assistant's Response with Predictive Text

Alex starts a chatbot conversation in the now-customary way, asking for a revision according to her goals 1. She now sees the assistant's response being formed—but instead of seeing the assistant type its response, Alex sees an editable text box, which starts empty except for the now-familiar buttons of predictive text.

Alex starts by ignoring the prediction buttons because she realizes it would be clearer to start with "The book", so she starts by typing that phrase. Afterwards the predictions give the title of the book, followed by the author, which Alex readily accepts with a few taps. After that, the top 3 suggestions are "revolutionized", "presents", and "is"; she take "presents", an active verb that avoids exaggeration. The next suggestions are "a revolutionary", "an innovative", and "a groundbreaking", which exhibit the same problem of exaggeration as before. These suggestions were probably due to Alex's prompt of "engaging", but the vacuous exaggeration of the suggestions indicates to Alex that she needs to consider what exactly *should be* engaging about this introduction. So she pauses to read the rest

of the article and concludes that the most important aspect is that the book provides a structured approach to thinking in individual and group settings. She needed to type "a structured", but then the predictions offered acceptable remaining words with only a bit of guidance: "approach to thinking, both individually and collectively."

Takeaways This interaction leverages the familiarity of the predictive-text interaction that is ubiquitous on smartphones, but the simple extension of this familiar interface to the context of typing the assistant's response to a revision request yields several unique kinds of uses:

- The system sometimes helps with routine tasks, like typing a book name (functioning like an adaptive copy-and-paste).
- The same interaction can suggest alternative wordings for phrases, using the natural 3– or 5–option button interface.
- Unlike a chat interface, the writer can exert granular, just-in-time control over the system.
- Some suggestions can even be provocative, leading the writer to pause and think more about what they wanted to say.

The prototype shows short phrases in prediction buttons, inspired by Arnold et al. (2016); next-phrase suggestions can shape writer thinking more than individual words even when not used directly (Bhat et al., 2023; Arnold et al., 2018; Jakesch et al., 2023).

3.2 Highlighting Edit Opportunities

Figure 2 shows a different interface with the same source text and prompt. This interface shows Alex's document with highlights in places where Alex might consider making edits to enact the revision goal that she has just specified. Hovering over an opportunity highlight shows a provocative clue of what an edit there might look like. Alex notices that "and" is highlighted; reading the phrase she notices that the phrase ("and the associated idea of parallel thinking") is not well connected to the main thought of the paragraph and decides to seek an alternative. Hovering the "and" reveals "is", suggesting that the next phrase could simply describe the book itself more (e.g., "is a guide for…") or perhaps state something concrete about its impact (e.g., "is the top-cited book on…"). Reading the rest of the paragraph and article, Alex decides to go with the description strategy, but chooses a different word: "describes a process for groups to plan thinking…". She makes this edit in the document and the opportunity highlights update to suggest other potential edits. She notices that the word "detailed" doesn't quite fit with how she understands the book; even though it is not highlighted, she hovers over it and sees an alternative, "structured", which seems more accurate.

Takeaways

- Alex retained full control over their document; all of the words are her own.
- In contrast to editing systems like Grammarly, Alex also had detailed control (via the prompt) over what sort of edit opportunities they wanted to see.
- The interaction allowed Alex to explore alternative choices: every word offered an alternative, even those not highlighted.
- The words shown in edit opportunities were sometimes substitutions but often instead offered a different semantic or grammatical direction that could be taken.
- It is still possible for the result to be entirely AI-generated text, but that would require the writer iteratively inspecting and applying every suggested change.

4 Discussion

So far these interaction designs have only been evaluated informally; empirical studies with writers are needed to determine how interaction-required suggestion interfaces affect writers' sense of ownership, control, and awareness of alternatives. Anecdotally, however (from use by the authors and a few others), both have been useful in low-level editing (trimming and clarifying wording), the predictive-text interface has been helpful for initial drafting (e.g., based on an outline), but neither are useful for larger-scale revision because they focus attention on localized choices; other tools are needed to address those needs (e.g., Dang et al. (2022); Benharrak et al. (2024); Kim et al. (2024)).

Although we described a case study in revision, predictive text could be used in any assistant response. We are particularly curious about how it might have different effects across different types

of tasks: open-ended tasks such as ideation, analytical tasks such as review generation, and close-ended tasks such as refactoring code.

The straightforward application of predictive text to typing the assistant's response, as we propose in Section 3.1, presents opportunities to increase cognitive engagement and control over the status quo of accepting complete generated responses. Yet it is still possible to use the chatbot's words uncritically by accepting suggestions rapidly. (Should the interface be designed to allow larger-block acceptance?) And even cognitive engagement with the suggestions could still lead to a reduced sense of ownership over the result (Lehmann et al., 2022) and influence on human opinions (Arnold et al., 2018; Jakesch et al., 2023). Additional exploration of the interaction design (e.g., how alternatives are visualized and navigated) is needed.

The additional control afforded by predictive text (effectively prefilling the assistant's response) affords some additional risks for users to jailbreak the model (Andriushchenko et al., 2024). However, since prefilling is part of many commercial LLM APIs, we doubt that this interaction design presents significant marginal risk.

Predictive text can be viewed as an interactive visualization of high-probability local alternatives within a sequence of categorical choices (e.g., Figure 1 shows two-token predictions to provide awareness of where each suggestion could be going.[1] From this perspective a wide range of interactive visualization techniques are possible, such as the Dasher text entry system (Ward et al., 2000) (which may have accessibility benefits as well). Design dimensions of these visualizations include the granularity of suggestions (words, phrases, or larger units such as copy-pasted text from a writer's other drafts) and how interacting with the suggestion affects the surrounding text. The effects of these design decisions might vary by stage of the writing process.

The opportunity highlighting interface explored an extreme design position of being minimally prescriptive in AI help, but relaxing that extreme could yield a range of alternative interaction designs. For example, it could incorporate an interactive visualization where the writer could navigate through contextual alternatives at any point.

[1]We plan to implement the phrase preview interaction of Arnold et al. (2016) to enable writers to see larger phrases without having to use all of them.

Both of these models presuppose autoregressive (left-to-right) language modeling, but additional types of interaction might be enabled by emerging model types based on out-of-order modeling or diffusion LLMs (Sahoo et al., 2024).

Although prior work has explored the effects of generating different kinds of content with LLMs on writer reactions (Benharrak et al., 2024; Kim et al., 2024; Zhou and Sterman, 2024), this work keeps the task for the LLM unchanged and explores the kinds of interactions that people can have with the inference process.

Interaction-required suggestions are a source of rich feedback data for reward-based language model training and personalization. Unlike static documents, the interaction logs with a conversational predictive text system would include what suggestions were made but not taken, providing a fine-grained human feedback signal. These feedback signals can be used for updating a language model (Wu et al., 2023; Arnold et al., 2017).

Conclusion With continuously increasing capabilities of LLMs, the difference between augmenting and replacing human thinking is a question not of system capabilities but of interaction design. The interaction-required approaches we've presented demonstrate how small shifts in interface design can fundamentally change the nature of human-AI partnership in writing. By prioritizing cognitive engagement, enabling granular control, and revealing the landscape of possibilities, we can design AI writing interfaces that help us think not *less* but *better*—maintaining human agency while still benefiting from AI capabilities.

Acknowledgments

We thank Hannah Yoo, Jason Chew, Juyeong Kim, Heonjae Kwon, Ray Flanagan, and the anonymous reviewers for their input and feedback. This work is supported by NSF CRII award 224614.

References

Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. 2024. Jailbreaking Leading Safety-Aligned LLMs with Simple Adaptive Attacks. In *The Thirteenth International Conference on Learning Representations*.

Kenneth C. Arnold, Kai-Wei Chang, and Adam T. Kalai. 2017. Counterfactual Language Model Adaptation for Suggesting Phrases. In *Proceedings of the Eighth International Joint Conference on Natural Language*

Processing (Volume 2: Short Papers), pages 49–54, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Kenneth C Arnold, Krysta Chauncey, and Krzysztof Z. Gajos. 2018. Sentiment Bias in Predictive Text Recommendations Results in Biased Writing. In *Graphics Interface 2018*, pages 8–11, Toronto, Ontario, Canada.

Kenneth C Arnold, Krzysztof Z. Gajos, and Adam T. Kalai. 2016. On Suggesting Phrases vs. Predicting Words for Mobile Text Composition. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16*, pages 603–608.

Karim Benharrak, Tim Zindulka, Florian Lehmann, Hendrik Heuer, and Daniel Buschek. 2024. Writer-Defined AI Personas for On-Demand Feedback Generation. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI '24, pages 1–18, New York, NY, USA. Association for Computing Machinery.

Advait Bhat, Saaket Agashe, Parth Oberoi, Niharika Mohile, Ravi Jangir, and Anirudha Joshi. 2023. Interacting with Next-Phrase Suggestions: How Suggestion Systems Aid and Influence the Cognitive Processes of Writing. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, IUI '23, pages 436–452, New York, NY, USA. Association for Computing Machinery.

Oloff C. Biermann, Ning F. Ma, and Dongwook Yoon. 2022. From Tool to Companion: Storywriters Want AI Writers to Respect Their Personal Values and Writing Strategies. In *Designing Interactive Systems Conference*, DIS '22, pages 1209–1227, New York, NY, USA. Association for Computing Machinery.

Arghavan Moradi Dakhel, Vahid Majdinasab, Amin Nikanjam, Foutse Khomh, Michel C. Desmarais, Zhen Ming, and Jiang. 2023. GitHub Copilot AI pair programmer: Asset or Liability? *Preprint*, arXiv:2206.15331.

Hai Dang, Karim Benharrak, Florian Lehmann, and Daniel Buschek. 2022. Beyond Text Generation: Supporting Writers With Continuous Automatic Text Summaries. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, UIST '22, pages 1–13, New York, NY, USA. Association for Computing Machinery.

Teresa Datta and John P. Dickerson. 2023. Who's Thinking? A Push for Human-Centered Evaluation of LLMs using the XAI Playbook. *Preprint*, arXiv:2303.06223.

Daniela Fernandes, Steeven Villa, Salla Nicholls, Otso Haavisto, Daniel Buschek, Albrecht Schmidt, Thomas Kosch, Chenxinran Shen, and Robin Welsch. 2025. Performance and Metacognition Disconnect when Reasoning in Human-AI Interaction. *Preprint*, arXiv:2409.16708.

Krzysztof Z. Gajos and Lena Mamykina. 2022. Do People Engage Cognitively with AI? Impact of AI Assistance on Incidental Learning. In *Proceedings of the 27th International Conference on Intelligent User Interfaces*, IUI '22, pages 794–806, New York, NY, USA. Association for Computing Machinery.

Katy Ilonka Gero and Lydia B. Chilton. 2019. How a stylistic, machine-generated thesaurus impacts a writer's process. In *Proceedings of the 2019 on Creativity and Cognition*, C&C '19, pages 597–603, New York, NY, USA. Association for Computing Machinery.

Angel Hsing-Chi Hwang, Q. Vera Liao, Su Lin Blodgett, Alexandra Olteanu, and Adam Trischler. 2024. "It was 80% me, 20% AI": Seeking Authenticity in Co-Writing with Large Language Models. *Preprint*, arXiv:2411.13032.

Maurice Jakesch, Advait Bhat, Daniel Buschek, Lior Zalmanson, and Mor Naaman. 2023. Co-Writing with Opinionated Language Models Affects Users' Views. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, pages 1–15, New York, NY, USA. Association for Computing Machinery.

Jiho Kim, Ray C. Flanagan, Noelle E. Haviland, ZeAi Sun, Souad N. Yakubu, Edom A. Maru, and Kenneth C. Arnold. 2024. Towards Full Authorship with AI: Supporting Revision with AI-Generated Views. In *Joint Proceedings of the ACM IUI 2024 Workshops*, volume 3660 of *CEUR Workshop Proceedings*, Greenville, South Carolina, USA. CEUR-WS.org.

Florian Lehmann, Niklas Markert, Hai Dang, and Daniel Buschek. 2022. Suggestion Lists vs. Continuous Generation: Interaction Design for Writing with Generative Models on Mobile Devices Affect Text Length, Wording and Perceived Authorship. In *Mensch Und Computer 2022*, pages 192–208.

Laria Reynolds and Kyle McDonell. 2021. Multiversal views on language models. In *2nd Workshop on Human-AI Co-Creation with Generative Models - HAI-GEN 2021*, volume 2903. CEUR.

Mohi Reza, Nathan Laundry, Ilya Musabirov, Peter Dushniku, Zhi Yuan "Michael" Yu, Kashish Mittal, Tovi Grossman, Michael Liut, Anastasia Kuzminykh, and Joseph Jay Williams. 2023. ABScribe: Rapid Exploration of Multiple Writing Variations in Human-AI Co-Writing Tasks using Large Language Models. *Preprint*, arXiv:2310.00117.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander Rush, and Volodymyr Kuleshov. 2024. Simple and Effective Masked Diffusion Language Models. *Preprint*, arXiv:2406.07524.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter

Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju-yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Cogan, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. 2024. Gemma 2: Improving Open Language Models at a Practical Size. *Preprint*, arXiv:2408.00118.

The Hugging Face Team. 2025. Hugging Face Transformers. https://huggingface.co/docs/transformers/index.

David J. Ward, Alan F. Blackwell, and David J. C. MacKay. 2000. Dasher—a data entry interface using continuous gestures and language models. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, UIST '00, pages 129–137, New York, NY, USA. Association for Computing Machinery.

Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A. Smith, Mari Ostendorf, and Hannaneh Hajishirzi. 2023. Fine-Grained Human Feedback Gives Better Rewards for Language Model Training. *Preprint*, arXiv:2306.01693.

David Zhou and Sarah Sterman. 2024. Ai.llude: Investigating Rewriting AI-Generated Text to Support Creative Expression. In *Proceedings of the 16th Conference on Creativity & Cognition*, C&C '24, pages 241–254, New York, NY, USA. Association for Computing Machinery.

Appendix

Implementation Details

The prototypes described here were implemented using a Streamlit frontend and a backend using the Hugging Face Transformers library (Team, 2025). Full source code and demo is available at `https://huggingface.co/spaces/CalvinU/writing-prototypes`.

Both of these systems rely on language model functionality that is not typically exposed in efficient ways in commercial APIs[2], but are straightforward to implement when given direct access to the forward pass of the model, which computes next-token distributions for all tokens in the context (including both "user" and "assistant" messages). The implementation in our demo uses the Gemma 2 9B model released by Google (Team et al., 2024).

The predictive text interface first computes the top-k (e.g., 3 or 5) next tokens, then constructs a short phrase (in the demo, a single additional token) by greedy generation from each of those options. With careful management of the key-value cache, this generation readily completes at interactive speed on commodity hardware. Predictive-text coding systems like GitHub Copilot served as informal prototypes of this interaction (since instructions can be entered as code comments), but they did not reveal the landscape of possibilities

[2]For example, prompt logprobs, needed for highlighting, was part of the OpenAI text completions API but was never added to the chat completions API

(see section 2 on Design Principles) in the way that smartphone keyboards and our system do.

The highlighting interface constructs a pseudo-conversation by where the user message is the revision prompt concatenated with the original document and the assistant message is the original document repeated unchanged. Rather than generate additional tokens, we simply compute the next-token distributions for all tokens in the "assistant" message corresponding to the user's document. The frontend highlights the tokens where the model gives a higher score to a token other than the one in the original document. Mouseover hovers show an alternative token; for tokens where the argmax prediction matched the original document (which are typically the majority of tokens), the hover shows the 2nd highest-scored option.