WORDFLOW: Social Prompt Engineering for Large Language Models

Zijie J. Wang **Aishwarya Chakravarthy** **David Munechika** **Duen Horng Chau**
College of Computing, Georgia Institute of Technology
{jayw, achakrav6, david.munechika, polo}@gatech.edu

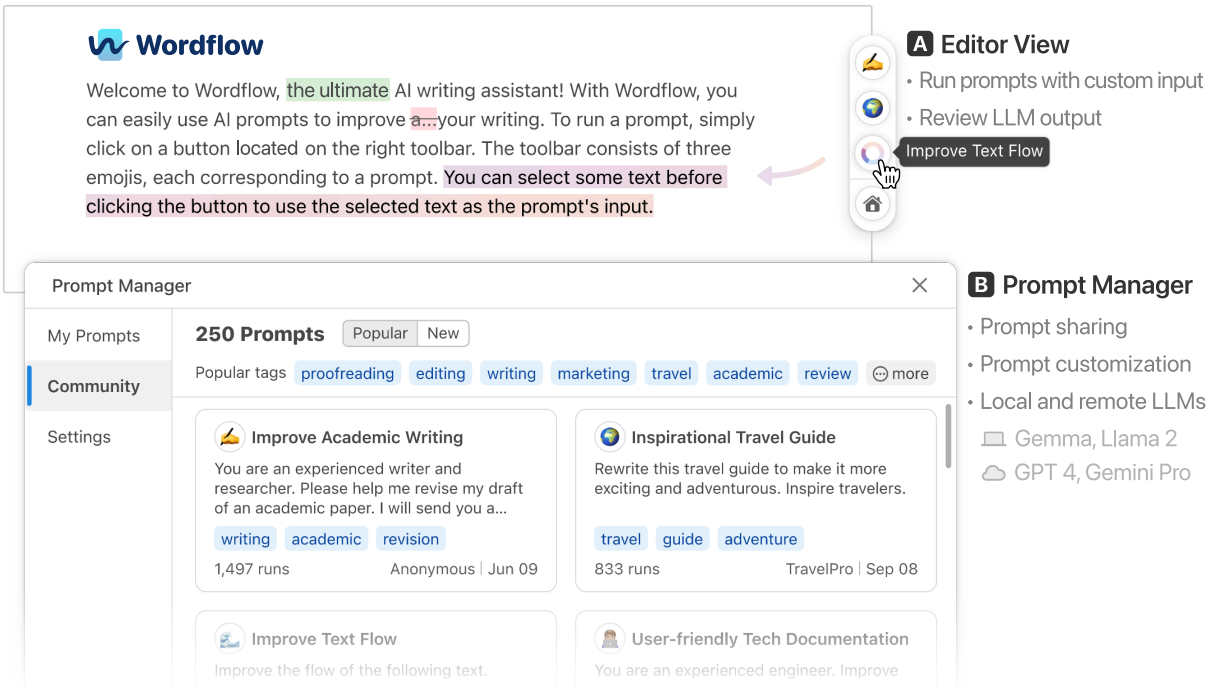


Fig. 1: WORDFLOW is an open-source social prompt engineering tool to help everyday users create, run, share, and discover prompts for large language models (LLMs). **(A) The *Editor View*** offers an easy-to-use text editing interface, allowing users to run an LLM prompt using the selected text as input by simply clicking on a button and examine the changes made by LLMs. **(B) The *Prompt Manager*** enables users to edit and curate prompts, adjust LLM settings, and share their prompts with the community.

Abstract

Large language models (LLMs) require well-crafted prompts for effective use. Prompt engineering, the process of designing prompts, is challenging, particularly for non-experts who are less familiar with AI technologies. While researchers have proposed techniques and tools to assist LLM users in prompt design, these works primarily target AI application developers rather than non-experts. To address this research gap, we propose social prompt engineering, a novel paradigm that leverages social computing techniques to facilitate collaborative prompt design. To investigate social prompt engineering, we introduce WORDFLOW, an open-source and social text editor that enables everyday users to easily create, run, share, and discover LLM prompts. Additionally, by leveraging modern web technologies, WORDFLOW allows users to run LLMs locally and privately in their browsers. Two usage scenarios highlight how our tool's incorporation of social prompt engineering can enhance laypeople's interactions with LLMs. WORDFLOW is publicly accessible at https://poloclub.github.io/wordflow.

1 Introduction

Recently, there has been a surge in the popularity of large language models (LLMs) such as GPT-4 (OpenAI, 2023a), Gemini (Team et al., 2023), and Llama 2 (Touvron et al., 2023). These pre-trained artificial intelligence (AI) models demonstrate a diverse array of capabilities that are continually being discovered, including summarization, question-answering, creative writing, and translation (Bommasani et al., 2022). To instruct these general-purpose LLMs to perform specific tasks, users need to provide them with *prompts*—text instructions and examples of desired outputs (Brown

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), pages 42–50

et al., 2020). These prompts serve as background contexts and guides for LLMs to generate text that aligns with users' objectives. Prompting enables users to employ LLMs for various tasks with plain language; in fact, well-crafted prompts can make general-purpose LLMs outperform specialized AI models (Nori et al., 2023).

Designing effective prompts, known as *prompt engineering*, poses significant challenges for LLM users (Jiang et al., 2022). LLM users often rely on trial and error and employ unintuitive patterns, such as adding "think step by step" (Kojima et al., 2022) to their prompts, to successfully instruct LLMs. Prompt engineering, despite its name, is considered an art (Parameswaran et al., 2023) and is even compared to wizards learning "magic spells" (Willison et al., 2022). Prompt writers may not fully understand why certain prompts work, but they still add them to their "spell books." Furthermore, prompting is especially challenging for *non-AI-experts*, who are often confused about getting started and lack sufficient guidance and training on LLMs and prompting (Zamfirescu-Pereira et al., 2023).

To help users prompt LLMs, researchers propose instruction tuning (Chung et al., 2022) and reinforcement learning from human feedback (Ouyang et al., 2022) to align a model's output with users' intent. Prompting techniques (Brown et al., 2020) are introduced to improve LLMs' performance on complex tasks. Libraries and interactive tools have also been developed to streamline the prompt crafting process (e.g., Chase, 2022; Jiang et al., 2022). However, existing techniques and tools primarily cater to AI application developers who use LLMs to build AI applications (e.g., chatbot applications), overlooking non-expert users who use LLMs for everyday tasks (e.g., checking emails for grammar errors). To bridge this critical research gap, we propose *social prompt engineering*, a novel paradigm that leverages social computing techniques to facilitate collaborative prompt designs. **We contribute:**

- **WORDFLOW, the first social and customizable text editor** that empowers everyday users to create, run, share, and discover LLM prompts (Fig. 1). It features a direct manipulation text editing interface for applying LLM prompts to transform existing text, such as proofreading and translation, or generate new text, such as creative writing. Users can easily customize prompts and LLM settings, share prompts with the community, and copy community prompts (§ 3). Two usage scenarios highlight how WORDFLOW and social prompt engineering can enhance users' interactions with LLMs (§ 4). Finally, we discuss future research opportunities enabled by our system (§ 5).
- **An open-source[1], web-based implementation** that lowers the barrier for everyday users in designing effective prompts and applying LLMs to their daily tasks. By leveraging modern web technologies, such as WebGPU (MDN, 2023; team, 2023), our tool enables users to run cutting-edge LLMs locally without the need for dedicated backend servers or external LLM API services (§ 3.4). Additionally, we offer an open-source implementation to help future designers and researchers adopt WORDFLOW for exploring and developing future user interfaces for LLMs. To see a demo video of WORDFLOW, visit https://youtu.be/3dOcVuofGVo.

We hope our work will inspire the research and development of collaborative interfaces that help everyone more easily and effectively use LLMs.

2 Related Work

Addressing prompt engineering challenges. Researchers have proposed libraries such as LANGCHAIN (Chase, 2022), GUIDANCE (Lundberg et al., 2023), and OUTLINES (Willard and Louf, 2023) to help users write prompts programmatically and control the structure of an LLM's output. By formulating prompting as programming, researchers propose techniques that help users edit (Fiannaca et al., 2023) and unit test prompts (Strobelt, 2023). COPROMPT (Feng et al., 2023) introduces a collaborative editor for multiple programmers to write prompts simultaneously. AI prototyping tools like PROMPTMAKER (Jiang et al., 2022), GOOGLE AI STUDIO (Google, 2023), OPENAI PLAYGROUND (OpenAI, 2023b), and PARTYROCK (Amazon, 2023) allow users to rapidly write and run prompts.

By leveraging visual programming techniques, tools such as AI CHAINS (Wu et al., 2022), PROMPT SAPPER (Cheng et al., 2023), and CHAINFORGE (Arawjo et al., 2023) enable AI application developers to visually design and test complex prompts. Similarly, PROMPTIDE (Strobelt et al., 2022), PROMPTAID (Mishra et al., 2023), and PROMPTERATOR (Sučik et al., 2023) employ mixed-initiative and interactive visualization tech-

[1]Code: https://github.com/poloclub/wordflow

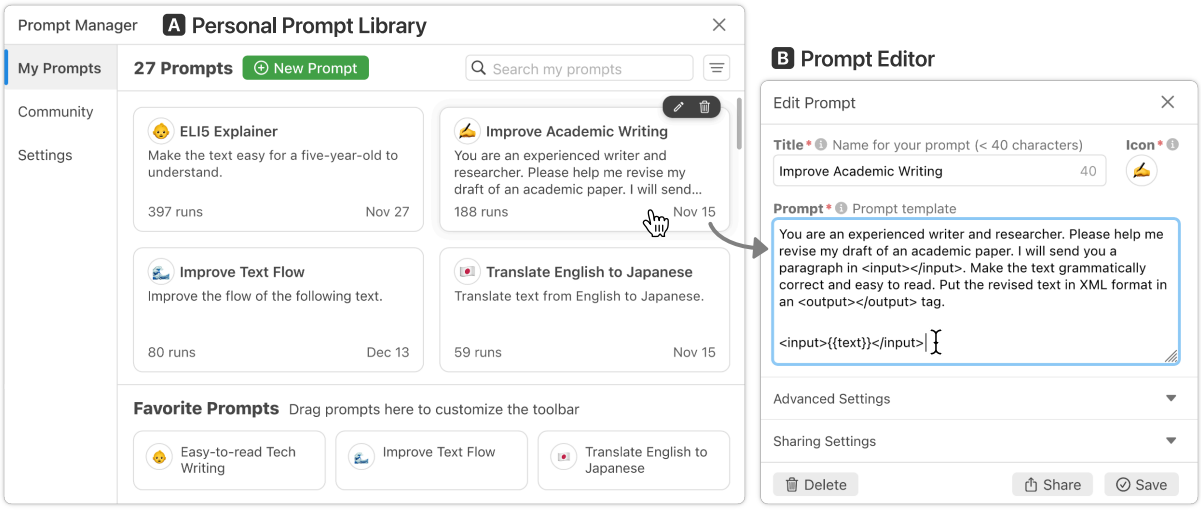


Fig. 2: Users can easily manage and customize their prompts in WORDFLOW. **(A) The *Personal Prompt Library*** provides an overview of local prompts, allowing users to search, sort, and customize the quick-action prompt toolbar in the *Editor View*. **(B) The *Prompt Editor***, activated by clicking a *Prompt Card*, employs progressive disclosure to help users modify prompt and configure output parsing rules, temperature, and sharing settings.

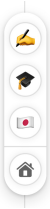
niques to help LLM users brainstorm and refine prompts. These existing tools function as IDEs that help *AI developers* craft prompts that will later be integrated into other applications. In contrast, WORDFLOW aims to serve as a runtime interface for *everyday users*, who act as both the prompt engineers and direct users of their prompts, and may not be well-versed in AI technologies.

Social prompt engineering. Online communities, including Promptstacks (Promptstacks, 2023), ChatGPT Prompt Genius (Reddit, 2023), and ShareGPT (Eccleston and Tey, 2022), serve as platforms for prompt creators to share tips, collaborate, and stay updated on AI advancements. User prompts from social media have been scraped to create prompt datasets for AI model development (Wang et al., 2023). Online prompt marketplaces, such as PromptBase (PromptBase, 2023), PromptHero (PromptHero, 2023) and ChatX (ChatX, 2023), have emerged to allow users to buy and sell prompts for generative models. Midjourney's Discord server (Holz, 2022) allows users to run and share prompts for text-to-image generative models, with dedicated sections for prompt critique and improvement (Oppenlaender, 2022). Building on the design of these communities, WORDFLOW provides an easy-to-use interface that unifies creating, running, sharing, and discovering LLM prompts. The most relevant related work is PROMPTSOURCE (Bach et al., 2022), an IDE for AI researchers and developers to write and share LLM prompts. PROMPTSOURCE targets AI experts using LLMs for natural language processing tasks on datasets (such as data annotation), and it requires users to provide a dataset. In comparison, WORDFLOW targets everyday users using LLMs for daily tasks, such as grammar checking, without the need to provide any dataset.

3 System Design & Implementation

WORDFLOW is an interactive tool that empowers everyday users to easily create, run, share, and discover LLM prompts. It tightly integrates four views: the *Editor View* (§ 3.1), where users can write text, run LLM prompts, and inspect changes made by LLMs; the *Personal Prompt Library* (§ 3.2), offering a prompt manager for creating and editing prompts locally; the *Community Prompt Hub* (§ 3.2), enabling users to explore and search for the latest and popular prompts shared by the community; and the *Setting Panel* for configuring remote or local LLMs (§ 3.4).

3.1 Editor View



When users open WORDFLOW in their browser or its mobile and desktop progressive web app, they are presented with the *Editor View* (Fig. 1A). This view shows a familiar text editor interface with a *Floating Toolbar* anchored on the right. Users can type or paste text into the editor. The *Floating Toolbar* consists of three prompt buttons and a home button (shown on the right). Each prompt button is represented by an emoji icon and corresponds to a prompt template. Users can click the prompt button to run its prompt using the current paragraph

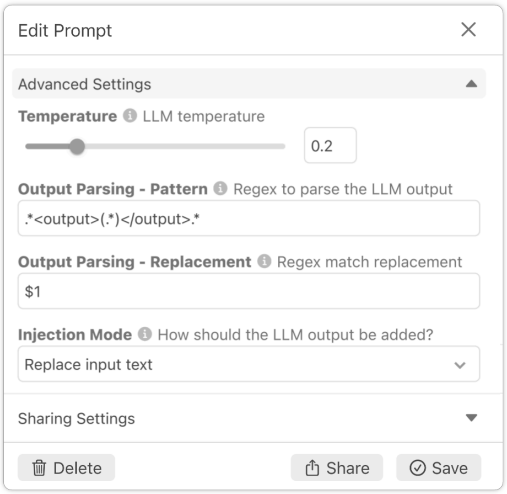


Fig. 3: Users can easily configure LLM temperatures and regex-baed output parsing in the *Prompt Editor*.

as the input text. If a user has selected some text, the selected text is used as the input for the prompt. Users can also click the home button to open a pop-up window that contains the *Personal Prompt Library* (§ 3.2, Fig. 2A), the *Community Prompt Hub* (§ 3.3, Fig. 1B), and the *Setting Panel* (Fig. 4).

Prompt input templating. In WORDFLOW, a prompt template includes pre-defined prefix text and a placeholder for the input text. For example, the prefix text can be "Improve the flow of the following text". The input placeholder in the template serves as a variable that will be substituted with the selected text from the editor. Inspired by popular prompting tools such as LANGCHAIN and PROMPTMAKER, our tool supports basic prompt templating. Users can include a special string `{{text}}` in their prompt template to represent the input placeholder (Fig. 2B), which will be replaced with the selected text from the editor before running the prompt. If the user does not include the string `{{text}}` in the template, the input text will be appended to the prompt template.

Prompt output parsing. To run users' prompts, WORDFLOW supports remote LLM API services, such as GPT 4 and Gemini API services provided by OpenAI and Google, as well as local open-source models, such as Llama 2 (Touvron et al., 2023) and Phi 2 (Abdin et al., 2023). Users can set their preferred models in the *Setting Panel* (Fig. 4). After receiving the output from the LLM API service or local model, the *Editor View* applies Myer's diffing algorithm (Fraser, 2012) to compare the output text with the input text. It then highlights the changes made by the LLM (e.g., addition, replacement, and deletion) using different text background colors (Fig. 1A). Users can click on the highlighted text to accept or reject the changes.

Inspired by LANGCHAIN, WORDFLOW allows users to add *optional* output parsing rules to a prompt by writing regular expression (regex) text (Fig. 3), which is useful for disregarding unrelated output text. For example, a user can prompt LLMs to structure the output in XML format (recommended by prompt engineering guidelines (Anthropic, 2023)), such as "Improve the flow of the following text. Put the rewritten text in an XML tag `<output></output>`". The user can then add a regex pattern `.*<output>(.*)</output>.*` and a replacement rule `$1` to parse the LLM's output before it is displayed in the *Editor View*.

3.2 Personal Prompt Library

After clicking the home button, users can open the *Personal Prompt Library* to manage their local prompts (Fig. 2A). This view organizes each prompt as a *Prompt Card*, allowing users to search and sort prompts based on name, recency, and run count. To change the prompts in the *Floating Toolbar* (§ 3.1), users can simply drag a *Prompt Card* into one of the three prompt slots located in the bottom row, each corresponding to a prompt button in the *Floating Toolbar*. To add or edit a prompt, users can click on the ⊕ New Prompt button or a *Prompt Card* to open the *Prompt Editor* (Fig. 2B). The *Prompt Editor* comprises three forms: basic prompt information (Fig. 2B), optional advanced settings (Fig. 3), and optional sharing settings. In the basic prompt information section, users can configure the title, icon, and prompt template. The advanced settings allow more experienced users to set the LLM temperature, output parsing rules, and insertion rules (Fig. 3). To share a prompt with the community, users can provide a description, tags, and recommended LLM models in the sharing settings, and then click on the Share button.

3.3 Community Prompt Hub

The *Community Prompt Hub* enables users to browse and search for prompts shared by WORDFLOW users (Fig. 1B). Each community prompt is represented as a *Prompt Card* and is associated with at least one tag. Users can filter prompts by clicking on a tag and can also sort prompts based on recency and popularity (i.e., the number of times they have been run). By clicking on a *Prompt Card*, users can access the *Prompt Viewer* (Fig. 5) to examine detailed information provided by the prompt

creator, including the title, description, prompt template, and recommended LLM models. Finally, users can click on the Add button to include a copy of the community prompt in their *Personal Prompt Library* (§ 3.2), where they can run the prompt, make further refinements, and potentially share it again with the community.

3.4 Open-source Implementation

We implement WORDFLOW as a progressive web app using Web Components and LIT Element (Google, 2015). Users can use it as a mobile or desktop app by saving it as a Safari Web App or a Chrome app. WORDFLOW allows users to run LLMs through remote API services, such as GPT 4 provided by OpenAI, or directly run open-source LLMs, such as Gemma (Team et al., 2024), Llama 2, Phi 2, and TinyLlama, in their browser (Fig. 4). We use Web LLM (team, 2023) and WebGPU to implement on-device LLM inference. In WORDFLOW, all local prompts are stored in the local persistent storage of the user's browser. To enable users to share community prompts, we use Amazon API Gateway and DynamoDB as a backend. Additionally, we provide a **Google Doc add-on** (Fig. 6) that allows Google Doc users to directly use WORDFLOW within their editor. We **open source** WORDFLOW as a collection of reusable interactive components that can be easily adopted by future researchers and developers in their interactive LLM projects.

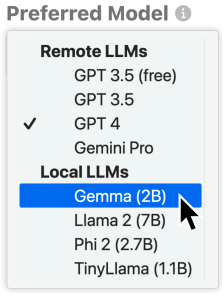


Fig. 4: WORDFLOW supports both remote and local LLMs.

4 Usage Scenarios

4.1 Improving Technical Writing

As a recently graduated junior software developer, Wade has been struggling with writing API documentation and system architecture descriptions. Specifically, Wade is unfamiliar with explaining technical concepts in simple language that can be easily understood by different colleagues such as developers, UX designers, and program managers. One day, Wade came across a forum thread where developers were sharing LLM prompts that had helped them improve their technical documentation writing. Wade had never thought about using LLM to assist him in his writing before. Intrigued,

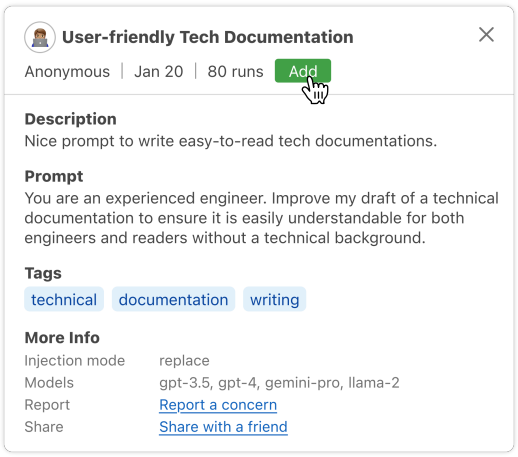


Fig. 5: The *Prompt Viewer* shows detailed information about a community prompt. Users can click a button to copy this prompt into their *Personal Prompt Library*.

he clicked on a WORDFLOW **prompt link** shared in a popular comment on the thread. The link opened WORDFLOW in a new tab, displaying the *Community Prompt Hub* along with a pop-up showing a community prompt (Fig. 5). Wade found the prompt and its description to be suitable for his writing tasks, so he clicked on the Add button to copy this community prompt to his local library.

Wade decided to try out this prompt to improve his writing. He opened the *Personal Prompt Library* and dragged the newly added prompt into one of the Favorite Prompts slots (Fig. 2A), and the prompt appeared in the *Floating Toolbar* in the *Editor View* (Fig. 1A). Wade copied a paragraph from the API documentation that he was working on. However, before clicking on the prompt button in the *Floating Toolbar*, Wade suddenly remembered that his company prohibits employees from using LLM services (e.g., ChatGPT and Bard) with work materials, as a measure to safeguard trade secrets and sensitive information.

Upon reviewing the documentation of WORDFLOW, Wade discovered that WORDFLOW supports running LLMs locally in browsers without sending any data to third-party services (e.g., OpenAI and Google). Therefore, he configured the LLM model to Llama 2, a local LLM model, in the *Setting Panel* before running the prompt on his writing. Then, he observed the changes made by the LLM model, which were highlighted in the *Editor View*, and found the new paragraph to be much easier to read. After using this prompt for a few days, Wade shared the prompt link on his company's mailing list, and more developers from his company began to use it to improve technical writing.

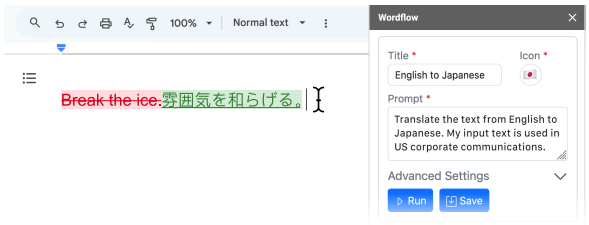


Fig. 6: Users can directly use WORDFLOW add-on to apply prompts to text within Google Doc documents.

4.2 Customizing Translation Styles

Ember, a senior manager in a US financial firm, recently encountered difficulties in communicating with her Japanese counterparts due to the absence of a translator. The use of traditional translation software has sometimes caused confusion among her Japanese colleagues. For example, the software translated the English idiom "break the ice" to "氷を砕く," which means "destroy the ice" instead of her intended meaning of "relieving tension when people interact for the first time."

Due to the recent popularity of LLMs, Ember decided to try using them to translate her documents from English to Japanese. As she writes in Google Docs, she explored the Google Doc Marketplace for an AI add-on and came across WORDFLOW. Upon installation, she opened the *Community Prompt Hub* (Fig. 1B) and selected the tag translation, which showed various popular translation prompts. She found a **prompt** titled "Translate English to Japanese."

After adding this prompt to her library, she tried to run it with the input "break the ice". However, WORDFLOW appended the incorrect translation "氷を砕く" to her document. Drawing from her previous experience interacting with ChatGPT, Ember decided to edit the prompt and provide additional instructions to guide the LLM model in considering her translation context. She opened the *Editor View* (Fig. 2B). and added a new sentence to the translation prompt: "My input text is used in US corporate communications" (Fig. 6 Right). Running the prompt again, WORDFLOW generated a more suitable translation "雰囲気を和らげる," which means "ease the atmosphere" (Fig. 6 Left). Ember back-translated the translation to English using her other translation software and felt more confident in continuing to use this prompt for future translations. Finally, to help other people who need to translate English to Japanese in business settings, she shared her **updated prompt** with the community by clicking on the Share button (Fig. 2B).

5 Future Work & Conclusion

In this work, we present social prompt engineering, a new paradigm that leverages social computing techniques to facilitate collaborative prompt design. To realize social prompt engineering, we design and develop WORDFLOW, an open-source and social text editor empowering users to easily create, run, share, and discover LLM prompts. Two usage scenarios highlight social prompt engineering and WORDFLOW can assist everyday users in interacting with LLMs. Reflecting on our design and development of this system, we discuss future research directions to help *everyone* use LLMs.

- **Usage log study.** Using WORDFLOW as a research instrument, we plan to conduct a usage log study to evaluate social prompt engineering and investigate (1) the effectiveness of social prompt engineering in helping everyday users craft prompts, and (2) everyday users' LLM use cases. We will examine the evolution of prompt editing and analyze community prompts to synthesize popular use cases and prompting patterns.
- **Fitting into user workflows.** Future tools like WORDFLOW can be seamlessly integrated into user workflows by being *in situ* and ubiquitous. With recent advancements in on-device machine learning, we see great potential for on-device LLMs, which allow users to avoid sending sensitive data to external services, reduce API costs, and use LLMs without network access.
- **Enhancing engagement in social prompt engineering.** There are great opportunities to enrich user interaction with LLMs using social computing techniques. To encourage user participation in prompt sharing, researchers can explore *intrinsic motivations*, such as designing an enjoyable social environment, and *extrinsic motivations*, such as virtual rewards and reputation systems.
- **Promoting responsible AI.** Social prompt engineering presents both opportunities and challenges for responsible AI. Platforms like WORDFLOW enable users to share prompting techniques to mitigate potential harms, but also run the risk of disseminating harmful prompts such as misinformation generators. In WORDFLOW, users can report harmful prompts, and we diligently monitor and moderate community prompts. Future researchers can explore social system designs that promote responsible prompting and develop methods to detect potentially harmful prompts.

6 Broader Impact

We propose social prompt engineering with good intentions—to help everyday users more easily and effectively use LLMs for their everyday tasks. In addition, we design and develop WORDFLOW, an open-source tool that is publicly accessible, to help everyday users easily create, run, share, and discover LLM prompts. However, bad actors might exploit our open platform to distribute prompts designed for harmful purposes. For example, they could share prompts that generate misinformation or content fostering divisive or extremist ideologies, exploiting WORDFLOW's reach to influence vulnerable and inexperienced audiences. To address and mitigate these potential harms, WORDFLOW enables users to report harmful prompts. We also actively monitor and moderate community prompts. Given that social prompt engineering and social prompting systems are still in their early stages, further research is needed to understand and address their potential harms.

Acknowledgements

This work was supported in part by an Apple Scholars in AI/ML PhD fellowship and J.P. Morgan PhD Fellowship. We are extremely grateful to Kaan Sancak, Jaemarie Solyst, Xinzhi Jiang for stress-testing our tool. We appreciate Tianqi Chen, Charlie Ruan, and Alex Cabrera for their suggestions and support for integrating on-device LLMs. We express our gratitude to Alex Bäuerle, Upol Ehsan, Muhammed Fatih Balin, Luis Morales-Navarro, Wesley Hanwen Deng, Young Wu, Giorgio Pini, Justin Blalock, Viraj Kacker, Seongmin Lee, Ben Hoover, Anthony Peng, Alec Helbling, Matthew Hull, Mansi Phute, Harsha Karanth, Pratham Mehta, Joanna Cheng, Lena Do, Smera Bhatia, Angelina Wang and Parul Pandey for their advocacy and valuable feedback.

References

Marah Abdin, Jyoti Aneja, Sebastien Bubeck, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, Suriya Gunasekar, Mojan Javaheripi, Piero Kauffmann, Yin Tat Lee, Yuanzhi Li, Anh Nguyen, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Michael Santacroce, Harkirat Singh Behl, Adam Taumann Kalai, Xin Wang, Rachel Ward, Philipp Witte, Cyril Zhang, and Yi Zhang. 2023. Phi-2: The surprising power of small language models.

Amazon. 2023. PartyRock: Everyone can build AI apps.

Anthropic. 2023. Introduction to prompt design.

Ian Arawjo, Chelse Swoopes, Priyan Vaithilingam, Martin Wattenberg, and Elena Glassman. 2023. ChainForge: A Visual Toolkit for Prompt Engineering and LLM Hypothesis Testing. *arXiv 2309.09128*.

Stephen Bach, Victor Sanh, Zheng Xin Yong, Albert Webson, Colin Raffel, Nihal V. Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, Zaid Alyafeai, Manan Dey, Andrea Santilli, Zhiqing Sun, Srulik Ben-david, Canwen Xu, Gunjan Chhablani, Han Wang, Jason Fries, Maged Alshaibani, Shanya Sharma, Urmish Thakker, Khalid Almubarak, Xiangru Tang, Dragomir Radev, Mike Tian-jian Jiang, and Alexander Rush. 2022. PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. 2022. On the Opportunities and Risks of Foundation Models. *arXiv 2108.07258*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child,

Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33.

Harrison Chase. 2022. LangChain: Building applications with LLMs through composability.

ChatX. 2023. ChatX: ChatGPT, DALL·E & Stable Diffusion prompt marketplace.

Yu Cheng, Jieshan Chen, Qing Huang, Zhenchang Xing, Xiwei Xu, and Qinghua Lu. 2023. Prompt Sapper: A LLM-Empowered Production Tool for Building AI Chains. *arXiv 2306.12028*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling Instruction-Finetuned Language Models. *arXiv 2210.11416*.

Dom Eccleston and Steven Tey. 2022. ShareGPT: Share your wildest ChatGPT conversations with one click.

Felicia Li Feng, Ryan Yen, Yuzhe You, Mingming Fan, Jian Zhao, and Zhicong Lu. 2023. CoPrompt: Supporting Prompt Sharing and Referring in Collaborative Natural Language Programming. *arXiv 2310.09235*.

Alexander J. Fiannaca, Chinmay Kulkarni, Carrie J Cai, and Michael Terry. 2023. Programming without a Programming Language: Challenges and Opportunities for Designing Developer Tools for Prompt Programming. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*.

Neil Fraser. 2012. Diff-match-patch: Hgh-performance library in multiple languages that manipulates plain text.

Google. 2015. Lit: Simple fast Web Components.

Google. 2023. Google Ai Studio: Prototype with Generative AI.

David Holz. 2022. Midjourney: Exploring New Mediums of Thought and Expanding the Imaginative Powers of the Human Species.

Ellen Jiang, Kristen Olson, Edwin Toh, Alejandra Molina, Aaron Donsbach, Michael Terry, and Carrie J Cai. 2022. PromptMaker: Prompt-based Prototyping with Large Language Models. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*.

Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large Language Models are Zero-Shot Reasoners. *Advances in Neural Information Processing Systems*, 35.

Scott Lundberg, Marco Tulio Ribeiro, and Harsha Nori. 2023. Guidance: A guidance language for controlling large language models. guidance-ai.

MDN. 2023. WebGPU API - Web APIs.

Aditi Mishra, Utkarsh Soni, Anjana Arunkumar, Jinbin Huang, Bum Chul Kwon, and Chris Bryan. 2023. PromptAid: Prompt Exploration, Perturbation, Testing and Iteration using Visual Analytics for Large Language Models. *arXiv 2304.01964*.

Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, Renqian Luo, Scott Mayer McKinney, Robert Osazuwa Ness, Hoifung Poon, Tao Qin, Naoto Usuyama, Chris White, and Eric Horvitz. 2023. Can Generalist Foundation Models Outcompete Special-Purpose Tuning? Case Study in Medicine. *arXiv 2311.16452*.

OpenAI. 2023a. GPT-4 Technical Report. *arXiv 2303.08774*.

OpenAI. 2023b. OpenAI Playground.

Jonas Oppenlaender. 2022. A Taxonomy of Prompt Modifiers for Text-To-Image Generation. *arXiv 2204.13988*.

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*.

Aditya G. Parameswaran, Shreya Shankar, Parth Asawa, Naman Jain, and Yujie Wang. 2023. Revisiting Prompt Engineering via Declarative Crowdsourcing. *arXiv 2308.03854*.

PromptBase. 2023. PromptBase: Prompt Marketplace: Midjourney, ChatGPT, DALL·E, Stable Diffusion & more.

PromptHero. 2023. PromptHero: Search prompts for Stable Diffusion, ChatGPT & Midjourney.

Promptstacks. 2023. Promptstacks: Your Prompt Engineering Community.

Reddit. 2023. R/ChatGPTPromptGenius.

Hendrik Strobelt. 2023. Prompt tester - quick prompt iterations for ad-hoc tasks.

Hendrik Strobelt, Albert Webson, Victor Sanh, Benjamin Hoover, Johanna Beyer, Hanspeter Pfister, and Alexander M. Rush. 2022. Interactive and Visual Prompt Engineering for Ad-hoc Task Adaptation With Large Language Models. *IEEE Transactions on Visualization and Computer Graphics*.

Samuel Sučik, Daniel Skala, Andrej Švec, Peter Hraška, and Marek Šuppa. 2023. Prompterator: Iterate efficiently towards more effective prompts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Pier Giuseppe Sessa, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L. Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu-hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. 2024. Gemma: Open Models Based on Gemini Research and Technology. *arXiv 2403.08295*.

MLC team. 2023. MLC-LLM.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv 2307.09288*.

Zijie J. Wang, Evan Montoya, David Munechika, Haoyang Yang, Benjamin Hoover, and Duen Horng Chau. 2023. DiffusionDB: A large-scale prompt gallery dataset for text-to-image generative models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Brandon T. Willard and Rémi Louf. 2023. Efficient Guided Generation for Large Language Models. *arXiv 2307.09702*.

Simon Willison, Adam Stacoviak, and Jerod Stacoviak. 2022. Stable Diffusion Breaks the Internet.

Tongshuang Wu, Michael Terry, and Carrie Jun Cai. 2022. AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts. In *CHI Conference on Human Factors in Computing Systems*.

J.D. Zamfirescu-Pereira, Richmond Y. Wong, Bjoern Hartmann, and Qian Yang. 2023. Why Johnny Can't Prompt: How Non-AI Experts Try (and Fail) to Design LLM Prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*.