AI and the Writer: How Language Models Support Creative Writers

Katy Ilonka Gero

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
under the Executive Committee
of the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2022

Abstract

AI and the Writer: How Language Models Support Creative Writers

Katy Ilonka Gero

Writing underlies a vast landscape of cultural artifacts, from poetry to journalism to scientific papers. While previously technology has been used to reduce the cognitive load of writing with accurate next word prediction, recent developments in natural language generation may prove able to go beyond predicting what we were going to write anyway, and give us new ideas relevant to a particular writing task. Such a proposal is quite new in the history of writing tools, and has so far proven illusory. Existing systems that address story continuation—presenting writers will options for the next sentence in their story—has continually found that suggested sentences are nonsensical, inconsistent with what's already written, or a deviation from the writer's intended direction. Thus, it's not understood if, and if so, how generative language technologies can support writers with complex writing tasks. I address this challenge by focusing on more specific goals than story continuation, and demonstrate that the methods I develop generate coherent, cogent suggestions that writers are able to use in a variety of settings and writing tasks. In this thesis, I consider writing tasks that are *constrained* by some external expectation, such as the logic of a metaphor or the details of a technical topic, but also require *creativity* to write a sentence or paragraph that is novel, surprising, and engaging to read. I present methods, embedded in systems, to support two challenging constrained, creative writing tasks. 'Metaphoria' aids in metaphor writing by generating metaphorical connections between two concepts. 'Sparks' aids in science writing by generating sentences that make a connection

between a technical topic and typical reader interests. These systems demonstrate that computation has the power to support constrained, creative tasks, and outline how they aid in inspiration, translation, and perspective. Furthermore, through a qualitative study with a range of creative writers, I uncover the social dynamics that modulate how writers respond to such generative writing support. Collectively, this work demonstrates new methods for using technology to support creative writers, and presents theoretical results that describe both how and why writers make use of such technologies.

Table of Contents

List of Figures

List of Tables

Acknowledgements

Chapter 1: Introduction

Creative writing is a keystone of human endeavors. From the journalism that sparks political action to the poetry recited between lovers, our ability to combine words into new and newly evocative texts remains a staple of human cultural activity. As we have entered the digital age via the ubiquity of computational devices, writing has come with us. We may write a text message on a smartphone or draft a report on a laptop; post a poem to Instagram or share a short story via a Google Doc. This shift to computing devices has opened up new affordances. Copy and paste, a labor intensive ordeal for pen and paper, has become a simple and common action in word processors. Collaboration, once requiring co-location or the time delay of sending and receiving a revised document, has become instantaneous and widespread. These changes in our collective writing practices have changed the shape of writing, the *how* of our writing, but they have not necessarily changed the *what*. In 'Track Changes: A Literary History of the Word Processor', historian Matthew G. Kirschenbaum documents the adoption of word processors by American novelists and finds, despite the varied opinions of how it will change literature (some think it will make prose more bland, others think more extravagant) that there's not much to distinguish a novel written longhand on a legal notepad from one written on a laptop. In his investigation, the *how* didn't seem to greatly impact the *what*.

Kirschenbaum looked at the word processor, but newer technologies—technologies that have changed rapidly over the last five years and likely will continue to change over the next five—may be on the precipice of changing the *what* of our writing. Advances in natural language processing, driven by increased compute power, better neural architectures, and an ever-increasing collection of digitized text, have created powerful *generative* technologies. These generative text systems seem fundamentally different from the affordances the personal computer originally allowed. Already our phones suggest the next word in our text message, and email clients attempt to finish our

sentences. In these cases the goal is typically to increase our typing speed (even if it does nudge our thoughts in a particular direction) but the technology behind these systems needn't just try to predict what we were going to write anyway. They may also be able to give us new ideas, help write a complicated sentence cleanly, or even become a set of external 'eyes', giving us feedback on what we've already written.

The history of computation as a creative writing tool is a long one. French and Romanian Dadaist writer Tristan Tzara, in 1920, would write poems on the spot by pulling words at random from a hat, a prescient and computational technique developed before the invention of computers, and a technique adopted again and again by various artists from David Bowie to William Burroughs. More recently, in 2019, the American dance-pop band YACHT produced an entire album in which all lyrics and melodies were generated by recurrent neural networks, although much curation and assembly was required. The difference between Tzara (and his contemporaries), working as an experimental writer pushing the limits of what we consider literature, and YACHT, attempting to write music in their own authentic style and voice, is worth considering. Experimental writing continues into modern times, making use of contemporary natural language generation to continue to interrogate the nature of language and writing. In this context, any technology may bear interesting artistic fruit, no matter its ability to understand or generate text in a human-like way. But for the average journalist or poet (or lyric-writer), the goal of writing is not to break the rules of their genre, but rather to express creatively within those rules. And it is this goal, of staying within the expected bounds of writing while writing exceptionally well, that new technologies may just now be able to support.

In this thesis, I consider writing tasks that are *constrained* by some external expectation, whether it be the logic of a metaphor or the details of a technical topic, but also require *creativity* to write a sentence or paragraph that is novel, surprising, and engaging to read. Recent work on story continuation—presenting writers with options for the next sentence in their story—has attempted to address a similar challenge, but has continually shown that writers find suggested sentences to be nonsensical, inconsistent with what has been already written, or a deviation from the writer's

intended direction [1, 2, 3]. This thesis presents methods that address more specific goals than story continuation, and demonstrates that these methods generate coherent, cogent suggestions that writers are able to use in a variety of settings and writing tasks.

The central question of this thesis is:

How can generative systems support writers in constrained, creative writing tasks?

This question is answered in several ways. First, the thesis presents methods for generating text to aid with constrained, creative writing tasks, and demonstrates that these methods both a) generate high-quality ideas and a) are used by writers who bring their own topic or intention to the task. Second, the thesis presents a model of the social dynamics involved in writers seeking computational or human support, based on interviews with creative writers.

1.1 Overview

This thesis is organized as follows:

Chapter 2 provides theoretical background for this work by summarizing relevant work from psychology, which has proposed various cognitive models of writing, and natural language processing, which has developed the technology used in this thesis. This chapter also gives a high-level overview of writing support tools to-date, which demonstrates that this thesis tackles the unsolved, challenging task of constrained, creative writing support.

Chapter 3 introduces a design space to characterize writing support tools, and presents a systematic literature review of such tools to formally demonstrate how the systems presented in this thesis take on previously unexplored parts of the design space.

Chapter 4 describes the design of Metaphoria, a system to support metaphor creation, chosen as a creative writing task constrained by the logics of the metaphor being written. This chapter reports on quantitative and qualitative studies demonstrating its success in adhering to the constraints of a provided seed metaphor while also encouraging creative, divergent outcomes for writers.

Chapter 5 describes the design of Sparks, a system to support writing explanations of scientific

topics, chosen as a task that requires more niche knowledge than Metaphoria (that of the technical topic being described) but still requiring creativity to make the explanation interesting to the average reader. This chapter reports on quantitative and qualitative studies demonstrating its success in adhering to the details of the technical topic while also promoting idea generation.

Chapter 6 reports on a qualitative study with creative writers and presents a taxonomy of the social dynamics involved in seeking support from a human or a computer. These dynamics modulate when and why a writer may seek help from a particular source. The work in this chapter is in direct response to the work on Metaphoria and Sparks, in which writers expressed a wide range of opinions of the systems, suggesting that the social dynamics of requesting support were an undocumented, confounding factor when evaluating writing support tools.

Chapter 7 concludes the thesis with a summary and discussion of the findings, and outlines new questions posed by these results.

1.2 Thesis Statement

This thesis demonstrates that interactive, generative writing systems can support writers in constrained, creative tasks by providing inspiration, translation, and perspective; furthermore, the social dynamics that underlie writers' desires for their writing, perception of support, and values in writerly interactions modulate writers' response to such systems.

1.3 Thesis Contributions

The contributions of this thesis are as follows:

1. A **design space** based on the cognitive process model of writing that characterizes systems that support creative writers.
2. A method for generating metaphorical connections between two concepts, exhibited in the system **Metaphoria**, and a study of this system demonstrating:
   (a) it is coherent to context and produces divergent outcomes for writers, and

(b) three ways professional poets make use of the system: to elicit ideas (inspiration), to overcome writer's block (translation), and to act as a new form (structural).

3. A method for generating sentences about a technical topic in order to aid in technical explanations, exhibited in the system **Sparks**, and a study of this system demonstrating:
   (a) three use cases of language model outputs—idea generation (inspiration), sentence construction (translation), and reader responses (perspective)—that correlate with distinct interaction patterns, and
   (b) that users prefer higher quality outputs within the outputs they see, but the overall quality of outputs seen does not correlate with perceived usefulness.
4. A qualitative **study of creative writers' attitudes** and behaviors with respect to generative writing tools, finding that the social dynamics that influence when and why they seek help from peers also influence their attitudes towards computational support.

Chapter 2: Background

2.1 Models of Writing

2.1.1 The Cognitive Process Model of Writing

Flower and Hayes' theory of the cognitive processes involved in writing [4] lay the groundwork for a plethora of research on the psychology of writing over the past four decades. This process model, developed through empirical think-aloud studies with writers, proposed that writing is best understood as a set of distinct thinking processes which are nonlinear and hierarchical. Figure 2.1 shows a schematic of the model, with the three main writing processes—planning, translating, and reviewing—highlighted in yellow. Planning includes generating ideas, setting goals, and organizing thoughts. Translating is the act of 'translating' ideas and thoughts onto words on the page; it is the literal act of writing.[1] Reviewing includes evaluating and revising what has been written.

When Flower and Hayes state that these processes are hierarchical, they mean that they can be called upon iteratively, being embedded within each other. For example, when a writer is constructing a sentence ('translating'), they may call in a compressed version of the entire writing process, for instance planning what the sentence will be about, selecting particular words, and evaluating the syntactical construction. Flower and Hayes' are also quick to note that these processes are not linear. While a common mantra is to 'plan, write, review,' in reality writers are making plans and reviewing what they have written all throughout the writing process.

Along with this process model, Flower and Hayes proposed that the act of writing is propelled by goals, which are created by the writer and grow in number as the writing progresses. These

[1] Flower and Hayes chose the word 'translating' over 'writing' to reduce confusion from overloading the word 'writing' with a more narrow, technical definition. In Computer Science, and especially Artificial Intelligence disciplines, 'translating' often refers to translating between languages, or even styles. Despite this, this thesis continues to use the word 'translating' in hopes of being less confusing than the word 'writing', which is used more colloquially to describe all parts of the writing process.

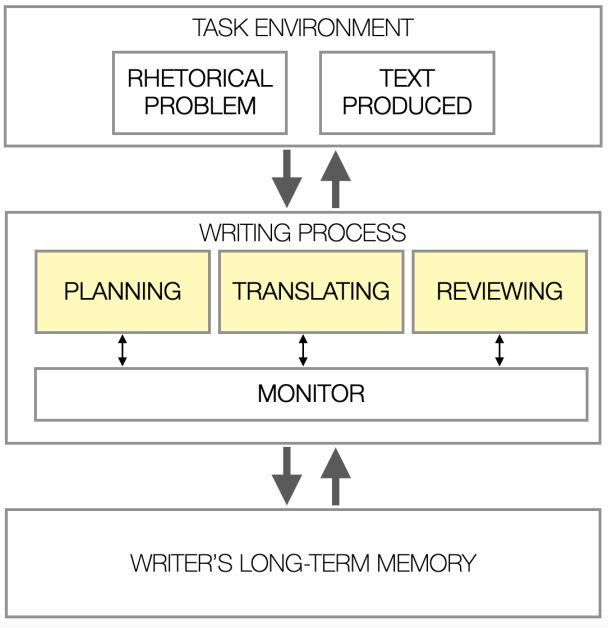


Figure 2.1: The cognitive process model for writing, as proposed by Flower and Hayes [4].

goals, which span in complexity and level of abstraction from 'appeal to a broad audience' to 'don't use that cliche', are what direct the writer to different processes. Thus we can model the writing process by considering the writer's goals and what processes they enlist to achieve these goals.

While this model has since been updated with an increase in complexity—Hayes adds much more detail to the long-term memory component, and adds components for working memory and the motivation and affect of the writer [5]—considering how goals propel the writing process remains a useful model. Writing has long been considered a mode of learning, as it is both a process and a product, which allows near-constant reflection on the ideas the writer is trying to express [6]. By considering a writer's shifting goals, writing researchers have understood why mature writers are able to learn from their writing [7]. Immature writers are often bogged down with low-level goals like sorting out syntactical issues or ensuring topical cohesion, which does not allow cognitive effort to be directed to more high-level goals. In contrast, mature writers are able to, when appropriate, set goals that require new knowledge to be generated. For instance, a mature writer may realize, when writing down an argument, that there is a logical gap and set

about bridging this gap, thus turning writing into a learning activity.

This thesis makes use of the cognitive process model of writing to be more precise in understanding what writing support tools help with. Proposing that a computational system helps with writing is vague and under-defined; instead one can propose what part of the writing process is supported, and then more accurately study if this is actually the case. The cognitive process model underlies all work presented in this thesis. It partially structures the design space presented in Chapter 3; it aids in the analysis of how writers use the systems presented in Chapters 4 and 5; and it shapes some of the results of the qualitative study presented in Chapter 6.

2.1.2 Writing as Creative Design

Research on creativity has proposed several models of the creative process. Csikszentmihalyi proposed a systems model of creativity, considered it a culturally embedded process that takes place within a *domain*, is validated by a *field*, and may be correlated with certain *individual* traits [8]. This work de-emphasized the individual and instead focused on the context in which creativity tends to take place. Others were more concerned with the act of creativity. Boden proposed three 'ways' to be creative: *combining* familiar ideas to create new ones, *exploring* unknown parts of a conceptual space, and *transforming* the bounds of a conceptual space to create new kinds of ideas [9].

While such work attempts to describe the occurrence of creative acts, it does little to help us understand the process that occurs during creative practice. Amabile proposed a process model of creativity that is remarkably parallel to the cognitive process model of writing: she described *preparation*, *generation*, and *validation* as a hierarchically embedded set of processes that reflect the creative process [10]. These processes align well with Flower and Hayes' model of writing, contains *planning* (preparation), *translation* (generation), and *reviewing* (validation) [4]. This alignment suggests that writing can be considered a creative practice.

Sharples suggested this when he considered the writer as a 'creative thinker and a designer of text' [11]. In particular he considers the creativity in writing occurring from the "mutually

promotive cycle of engagement and reflection", essentially the iterative process of design. He too points out parallels between the writing process and the creative design process, and argues that aligning writing with design emphasizes that writers, like designers, are users of tools.

2.2 Natural Language Generation

2.2.1 Introduction to Language Models

This thesis deals in writing support via the generation of text a writer may want to incorporate, either verbatim or conceptually, into their own writing. The core model for text generation in this thesis is the language model. A language model is any model that predicts the likelihood of a sequence of words. For instance, consider the following two sentences:

1. The squid swam in the sea.
2. The squid sea the swam in.

We want a language model that thinks sentence 1 has a higher probability than sentence 1, demonstrating that the language model predicts syntactically correct sentences are more likely than syntactically incorrect ones. However, a language model can also be used to compare semantics. Consider these sentences:

1. The squid swam in the sea.

2a. The squid wrote the thesis.

A language model that predicts sentence one is more likely than sentence 2 is reflecting some norms of language we may want captured, like squids are more likely to swim than write. Because a language model must be trained on a corpus of text, it always reflects the norms of the language it is trained on. A language model trained on a corpus with no mention of squids may predict both of these sentences have equal likelihood, or a model trained on children's books may predict that a squid writing is just as likely as a squid swimming.

Language models can be used to generate text by giving the model a prefix of text and having it calculate the likelihood of each word in its vocabulary as the next word. This probability distribution can be used to select the next word, and thus generate text continually by adding the selected next word to the prefix. A simple text generation picks the next word that has the highest likelihood, but alternative methods may conduct a beam search or stochastically sample from the probability distribution.

2.2.2 Neural Language Models

Language models can predict the likelihood of a sequence of tokens in a variety of ways. In this thesis, neural language models—language models that make their predictions using neural networks—are used as they have become the standard (and most successful) way to model language. There are several benefits of neural language models over past models. Neural language models allow for vector representation of tokens, resulting in learned representations that capture rich semantic information [12]. While past models such as n-gram models struggled to account for longer prefixes (i.e. more past text) without becoming exponentially larger, neural language models have introduced several methods to do so, for instance by using context vectors as in recurrent neural networks or long short-term memory networks, or by linearly increasing the size of the networks, as in transformer models [13].

In recent years, language models have been getting larger; that is they are being trained on more text and the models have more parameters [14, 15, 16]. These models, often with little to no adaptation, succeed in a variety of tasks, compared to prior smaller models typically trained for a particular purpose. Much resent work has been done on how to make the best use of these large, pre-trained models.

Despite the successes of these models, problems remain. Language models tend to output repetitive and vague responses [17, 18]. They have no model of the truth; they are learning correlations from large amounts of text and thus are able to generate falsehoods. Finally, it has been well-documented that these models can generate offensive language, have distributional biases,

and may copy text from the training data [19, 20].

2.2.3 Practical Issues Using Neural Language Models

Decoding Method There are several common ways of deconding from language models: greedy search, beam search, top-k sampling [17], and top-n sampling [21]. Different methods have different strengths and weaknesses. Greedy search, which selects the most likely word at each generation step, is rarely used for creative text generation as it tends to produce very generic responses (and rarely finds the most likely sequence of words). In contrast, beam search, which maintains a 'beam' of n possible outputs, can find more likely sequences an tends to produce high quality results [22]. When trying to generate multiple possible outputs for the same prompt, sampling methods, where words are sampled from the language model according to their likelihood, are often used. However this often decreases the coherence of the outputs, because very unlikely words can not be generated with some (albeit small) probability.

Prompt Engineering It has been shown that a well-selected prefix, or 'prompt', can dramatically increase the performance of a language model on a specific task [23]. A resulting line of research has looked at how to search for or train prompts, and Li et. al. provide an excellent survey of this emerging field [24]. A useful distinction made in the survey is between discrete prompts, which are natural language prompts that read like normal text, and continuous prompts, which operate over the vector space of the language model. Continuous prompts have outperformed discrete prompts for GPT3 and BERT in some settings, suggesting that continuous prompts may produce better outcomes even if natural language prompts are more intuitive [25]. However, these results are dependent upon many factors, for instance the directionality of the language model (unidirectional vs. bidirectional), the scale of the model, the method of selecting or training the prompt, the kind of training data utilized, and the type of downstream task [24].

2.3 Writing Support Tools

Writing support has a long history; dictionaries and thesauruses represent old, analogue tools that continue to be used by writers today. This section outputs a variety of computational approaches to writing support, demonstrating that generative support for constrained, creative writing remains an open and challenging problem.

2.3.1 Correctional Support

One of the early successes of computation was the development of spell-check [26, 27], which today exists as a standard auto-correct feature in most word processes. Grammar-checking then became an important area of research, especially to support language learners [28], and remains an active area of research today [29]. However, correctional-style support, while highly constrained by the standard rules of language (spelling, syntax) doesn't support the creation of new ideas or new ways of expressing ideas.

2.3.2 Text and Word Prediction

Text prediction was traditionally studied as an aid for people with disabilities (e.g. motor and speech impairments) [30]. More recently, they have been developed as general purpose writing support tools, as noted by the default suggestion of words when typing on a mobile keyboard. Writing support designed explicitly for messaging tasks such as email and texting tends to be much more constrained, as the explicit goal is to predict what the user would have written anyway [31]. For instance, Gmail's Smart Compose sentence completion and Smart Reply suggestions aim to "[cut] back on repetitive idiomatic writing" [32, 33]. Though it has been shown that even these suggestions can change what people write [34], these tools intend to suggest only text that the writer would have written anyway. Similar to correctional support, although highly constrained, these systems do not deal in ideation, planning, or creativity.

2.3.3 Providing Examples

Writing support tools have tackled more complex and abstract writing tasks by providing writers with examples. IntroAssist [35] supports writing help request emails by providing writers with a checklist and example emails. INJECT [36] supports journalists by providing a guided creative search bar, with templated concept cards to aid in retrieving relevant web content. CreativeHelp [37] uses case-based reasoning to retrieve relevant next sentences from a corpus of stories for someone writing their own story.

While these tasks are more creative than correctional-style support or word prediction, they rely on hand-crafted examples, web searches, or returning text written by others. A system that relies on existing text written by others will struggle to support unseen contexts with novel ideas. Additionally, writers may express worry about the source of the text, and may not feel comfortable using or being influenced by these sources.

2.3.4 Crowd-sourcing Support

Other work makes use of crowd workers to provide support to writers. Soylent [38] uses the idea of microtasks to produce high quality edits from crowd workers. Heteroglossia [39] allows writers to share snippets of their story draft with crowd workers and ask the crowd to provide follow-up ideas. Crowd ideation is a more general version of this task [40, 41]. While crowd workers may be able to generate new ideas for a variety of contexts, in the evaluation of these systems writers expressed difficulty working with strangers, and reported fear that it would raise copyright issues [39].

2.3.5 Generative Support

This thesis investigates the use of generative systems, where a system generates novel text for the writer. Generative systems can produce unique text for an endless number of unseen writing contexts. This it not a completely new idea. Roemmele's thesis work investigated using neural networks to generate next sentences in a story as a creative writing support tool [1]. Her work

is in several ways a precursor to the work presented in this thesis. She aimed to support writers doing a creative task—writing a story—that is constrained by what they have already written. She calls this task 'story continuation', and it was been adopted by others (e.g. [2, 3] as a challenging generative task.

However, in her evaluation of such systems, she found "the most common piece of feedback was that the suggestions were not coherent." Some suggestions were completely nonsensical, while others did not fit with where the story was going. This issue of failing to cohere to the constraints of the writer's goals has continued to plague researchers. Clark et al. [2], developing a similar system for story continuation, found that "all participants said that the suggestions were very random", with a majority of participants saying they would only use the system if the suggestions were better (and some saying they would not use it at all.)

I investigated this issue myself. In Calderwood et al. [3] we tested the use large language model GPT-2 as a story continuation system with professional novelists. Similar to Roemmele [1] and Clark et al. [2], we found that sentence continuation continually "deviate[d] from the direction [the writers] were taking in their writing."

This thesis addresses these challenges by supporting writers with more specific goals than story continuation. I develop custom text generation systems that address more specific goals, and demonstrate that these systems do provide coherent, relevant suggestions to writers. Furthermore, this thesis evaluates the ways in which writers make use of suggestions (since they are now useful and thus actually used by writers), and presents a taxonomy for understanding what modulates writers desires to use such systems.

Chapter 3: A Design Space for Writing Support Tools

This chapter draws on the cognitive process model of writing [4] to propose a design space for writing support tools. The design space gives designers and researchers a tool to better understand what a writing support tool is attempting to support, and identify gaps or opportunities in the field of writing support tools more generally. It provides a shared vocabulary, as well as common goals and methodologies.

To demonstrate the use of the design space, we perform a systematic literature review of research on writing support tools from the last five years (2017-2021). This shows areas of active research and under-served areas, as well as limitations of current technology to support different aspects of writing. We also use these papers to investigate how to best evaluate writing support tools.

The contributions of this section are:

- A design space for writing support tools, based on a cognitive process model of writing.
- A systematic literature review of writing support tools ($n_{papers} = 30$) from 2017-2021.
- A gap analysis highlighting opportunities for designing future writing support tools.
- A series of common evaluation methodologies for future work to draw on.

3.1 Related Work: Design Spaces

One way to synthesize a multitude of designs is to envision it in a 'design space', or a metaphorical laying out of designs according to some metrics or measures. MacLean et al. [42] describe design space *analysis* as an approach to representing design rationale. In this view, a design space places a design in a "space of possibilities" and uses this placement to explain why a design was chosen among all the various possibilities. This frames design spaces as a useful way of commu-

nicating with stakeholders. By explaining why a design was chosen, stakeholders can better sell, maintain, and otherwise interact with a product.

Woodbury and Burrow [43], addressing the growing popularity of design spaces in computational research, describe design space *exploration* as a way to consider design choices. Exploration also provides compelling model of design. Again, they propose representing designs in a meaningful way, and using the representation to explore the resulting 'space'.[1]

A popular and highly-cited example of a design space comes from wireless sensor networks. As the use of such networks increased globally, "it was very difficult to discuss specific application requirements, research directions, and challenges" [44]. The proposed solution was a sensor network design space: its various dimensions would be categorized in order to both understand the existing research as well as discover new designs and applications. One conclusion was that a small set of platforms could cover the majority of the design space, rather than requiring numerous, application-specific platforms.

This chapter introduces a design space both to think about what writing support tools currently do, and what we might want them to do in the future. In this sense it takes both MacLean's and Woodbury's view: the design space is both a way to talk about why existing tools are the way they are, as well as a way to design new ones.

3.2 Writing Goals Design Space

In their seminal paper on the cognitive processes involved in writing, Flower and Hayes [4] describe writing in the following way:

> *The act of composing itself is a goal-directed thinking process, guided by the writer's own growing network of goals.*

These writing goals may be large, like to write up the experiment metholodogy for an academic paper, or small, like to make a sentence sound more formal. They may be open-ended, like to come

[1] They also propose using a design space to build computer systems to aid in the design process but automating the exploration.

up with the name for a character, or quite limited, like to spell a word correctly. The goals may require imagining the reader, like to determine if a sentence is too confusing, or they may require diving deeper into what's already written, like to ensure a technical topic is discussed consistently throughout an article. Writing goals may start as external motivators—someone may ask the writer to write something—but as the writer writes, new writing goals are created by the writer themself and propel the writing process forward.

This chapter proposes using the idea of writing as *goal-directed thinking process* to structure a design space for writing support tools. Whether they are called support tools, assistants, co-creators or machines-in-the-loop, what unites these systems is that they take on goals inherent in the writing process.

The design space has two axes:

1. Which part of the writing process the system aims to support. Flower and Hayes, in their original model of writing, propose three components: planning, translating, and reviewing. These three components align with models of creativity, which often cite ideation, implementation, and evaluation [10]. In both cases the components are accessed iteratively, and often hierarchically. A writer may start with a high-level plan, and then in the act of translating or implementing the plan may create a smaller plan within it. Splitting up writing support tools into these processes helps us understand how, when, and why a writer may use a tool.

There can be some ambiguity in distinguishing between these processes. For instance, consider a tool that, upon request, completes a writer's sentence. This tool may be supporting translating, if the completion is intended to articulate what the writer already had in mind. Or it could be supporting planning, if the completion is intended to provide the writer with new ideas or directions for their writing. When annotating papers, we rely on how the researchers describe the tool, though we acknowledge that writers may use a tool in unexpected or unintended ways.[2]

2. The amount of constraint the goal has. A highly constrained goal has very few possible solutions, like when writing a technical definition. A lightly constrained goal has many possible

[2]An alternate approach is to rely on how writers describe their usage, but given that many papers did not include this in their evaluation, we would not have been able to annotate all papers using this method.

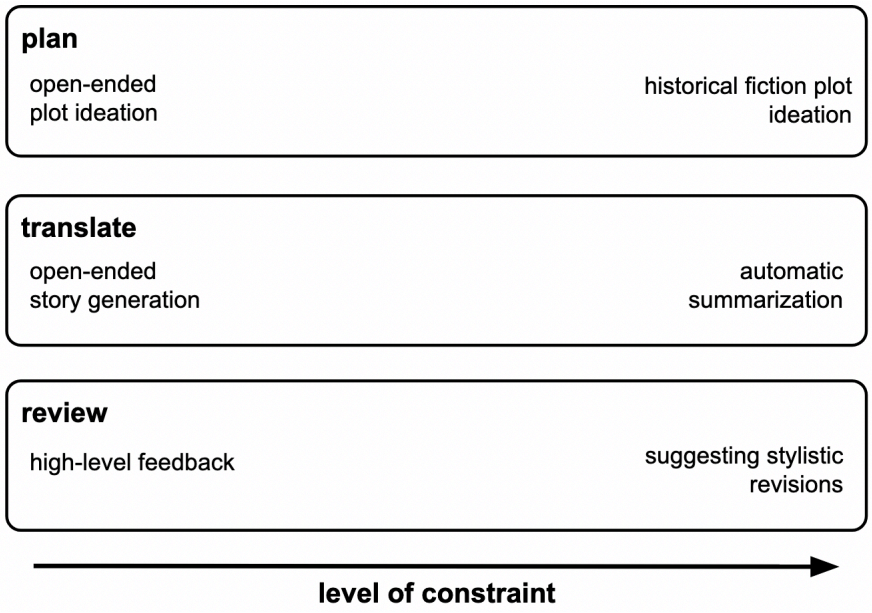


Figure 3.1: The writing goals design space is defined by the part of the writing process a tool wants to support and the level of constraint of the goal. This shows some example writing goals a tool may want to support.

solutions, like when describing a newly introduced fictional character. The amount of constraint gives us a measure of how particular the support must be to achieve the goal. This may be considered a measure of difficulty—writing a technical definition is very constrained, and supporting this writing task requires a high level of world understanding from a system—but constraint doesn't always imply difficulty. A writing goal may be very constrained, for instance make a particular sentence more positive, but the support may be fairly straightforward, like providing a list of positive words.

Figure 3.1 shows some hypothetical writing support tools in this design space, to better understand the space.

3.3 Literature Review

3.3.1 Methodology

We perform a preliminary, systematic literature review such that we can plot tools in the design space. This validates the utility of the design space and provides insights into the landscape of

writing support tools.

We design a query for searching the ACM Digital Library for relevant papers. Our goal for this query is to find as many relevant papers as possible, while minimizing the number of irrelevant papers needed to sort through. This proved more difficult than expected because search terms like 'writing' and 'support' are quite common in other subfields, like those studying memory architecture. We iterated on a query that returned many of the papers we expected to be included (such as [45] and [46]), while also returning less than 300 results, such that we could visually inspect them all. We chose to only look at papers from the last five years as we wanted to focus on where the field is currently going. We didn't require an average yearly download or number of citations, as done in other systematic reviews like [47], because we wanted to include very recent work that may not be well-distributed yet.

Our final query was:

> [[Abstract: writing] OR [Abstract: writer]] AND
>
> [[Abstract: interface] OR [Abstract: system] OR [Abstract: prototype] OR [Abstract: tool]] AND
>
> [[Abstract: assistant] OR [Abstract: support] OR [Abstract: tool]] AND
>
> [Publication Date: (01/01/2017 TO 12/31/2021)] AND
>
> [CCS 2012: Human-centered computing][3]

This query resulted in 216 items.

We then had to select which of these papers to include. First we had one researcher read the titles of all papers and perform a quick 'desk reject' on any papers that were clearly off topic.[4] After this, 77 papers remained. Of these papers, two researchers read all the abstracts and noted

[3]The results of the query can be found at the following url: `https://dl.acm.org/action/doSearch?fillQuickSearch=false&target=advanced&expand=dl&CCSAnd=60&AfterMonth=1&AfterYear=2017&BeforeMonth=12&BeforeYear=2021&AllField=Abstract%3A%28writing+OR+writer+OR+writers%29+AND+Abstract%3A%28interface+OR+system+OR+prototype+OR+tool%29+AND+Abstract%3A%28assistant+OR+support+OR+tool%29`

[4]For example, a paper with the title 'A Tool for Visualizing Classic Concurrency Problems' was rejected for clearly being about a different topic.

if they thought a paper should be included based on the inclusion criteria below. They did this separately, and then came together to discuss and resolve disagreements.

Our inclusion criteria was:

1. a conference or journal publication[5]
2. a contribution that presents or studies a tool that aids in the translation of ideas into text

Below are examples of types of papers that would or would not be included. We used these examples when determining which papers would be included.

- Some examples that **would not** be included: a general purpose productivity tool, where writing is an example use case; a study/analysis where the data analyzed is writing data; a study about writing-adjacent tools, like handwriting recognition; a tool that generates writing with little human interaction; a non-writing tool with a language interface; language learning tools.
- Some examples that **would** be included: a design fiction about a writing tool; a writing tool that has no evaluation; a writing tool that writes the first draft and then a human revises it; a study of a commercial writing tool; a tool that supports a very specific writing task; a tool that supports writing and something else (but is not a general purpose tool).

This resulted in 30 papers. Each paper was assigned a nickname which allowed for easier reference than the paper title or author list.

We then annotated the selected papers. Three members of the research team participated in the annotations. The selected papers were split up, and each paper was annotated by a single researchers. Some of these annotations were to allow us to plot tools in the design space, others were to align with [47], a systematic review of creativity support tools, and still others were to quantify the type of contribution. The results of our annotations can be found at `https://github.com/kgero/writing-support-tools-2022`.

[5] i.e. not a course description, workshop proceedings, etc.

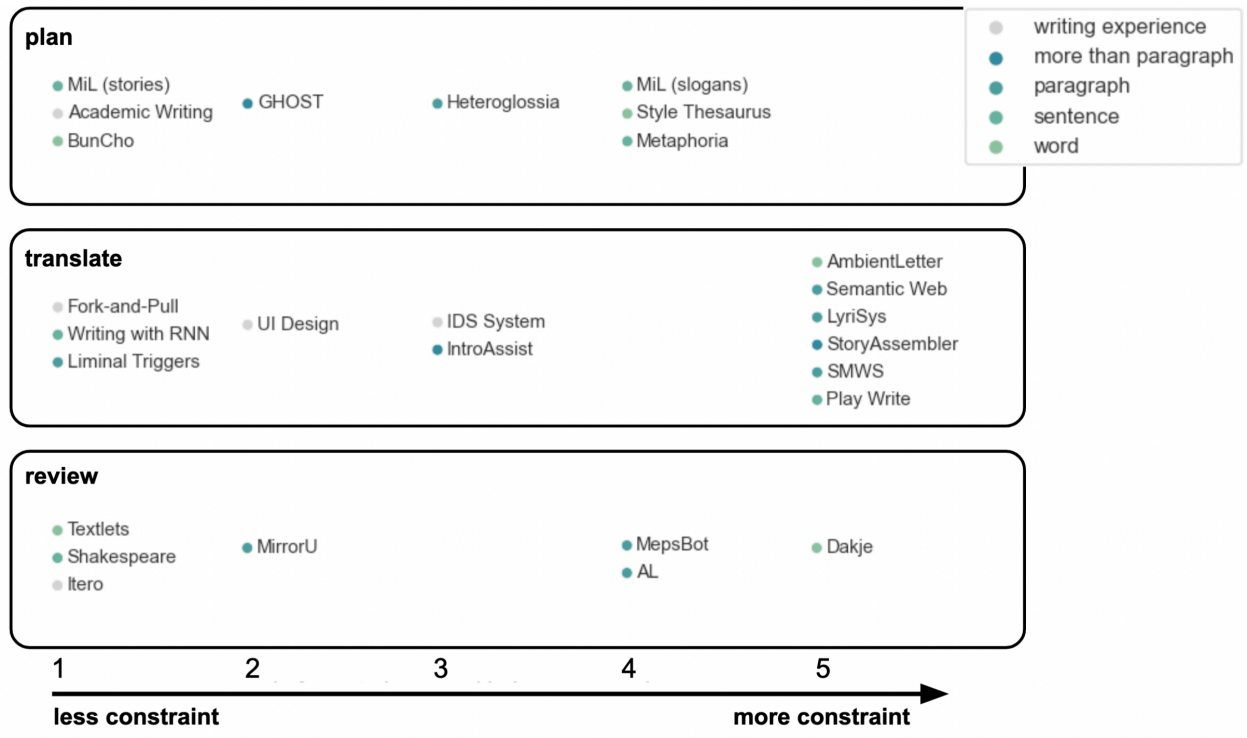


Figure 3.2: Twenty-seven writing support tools plotted in the writing goals design space. We can see that highly constrained planning and reviewing are under-explored areas.

3.3.2 Results and Analysis

In this section we consider how tools are distributed in the design space, which looks at the type of goal the tool supported, and how constrained that goal is. The 30 papers represented 33 systems, with some papers presenting multiple systems.[6] Three papers studied tools that supported all parts of the writing process: Writing Together [49] studied Google Docs, Writing on Github [50] studied GitHub, and Literary Style [51] presented an early stage exploratory tool. We exclude these because it is difficult to locate them in a single part of the design space; future work may consider how tools can be distributed across multiple parts of the design space. Excluding these, we are left with 27 systems to analyze in this section.

Figure 3.2 shows all tools in the writing goals design space. We color them by the size of the goal being supported. We see most parts of the design space covered, with tools in all three parts

[6]UI Design [48] studied four systems, but since they were all very similar, for this section we consider them to be a single system (as they would be in the same part of the design space anyway).

of the writing process and spanning many different levels of constraint. The papers also operate on all different sizes of writing goals.[7]

The design space shows that planning and reviewing lack work on highly constrained support, motivating an area for future work. There are no planning tools with constraint=5, and just one reviewing tool with constraint=5, compared to the six translating tools with constraint=5. Of the three planning tools with constraint=4, one is presented in this thesis (Metaphoria, Chapter 4), and another is my own work outside the scope of this thesis (Style Thesaurus, an early stage exploration of a customized thesaurus system [52]). This more formally demonstrates that constrained ideation tasks, which would fall in the planning category, have yet to be addressed by the research community.

As the constraint for the goal increases, tools tend to support narrower and more structured writing tasks. MiL (stories) [53] and BunCho [54] (both constraint=1) support any kind of story writing, while MiL (slogans) [53] and Metaphoria [55] (both constraint=4) support slogan and metaphor writing respectively, which have rules and syntactic structures to guide the generation process. Reviewing similarly sees this move towards the niche as constraint increases. Textlets [56] (constraint=1) is a general purpose reviewing tool based on a sophisticated usage of the 'find' command. In contrast, MepsBot [57] (constraint=4) focuses on comments in online mental health forums and Dajke [58] (constraint=5) is about adjusting the reading level of Tibetan learning material.

Does a highly constrained writing goal need to be niche or highly structured? It may be that language technologies have not yet been capable of supporting more general purpose but still highly constrained writing goals. For instance, brainstorming often happens at multiple points throughout a creative process, with later brainstorming being more constrained by previous choices. Early stage brainstorming may be easier to support because there are less constraints needed to get right.

[7] Five at the level of words, six at sentences, eight at paragraphs, three at more than the paragraph, and five on the writing experience.

An area new technologies could explore is later-stage brainstorming, which could be quite general purpose—input any piece of writing and a brainstorming prompt—but still lie in the highly constrained planning part of the design space.

The design space shows that highly constrained support for translation is well studied. These systems tend to support highly structured writing tasks. AmbientLetter [59] supports spell-checking while writing on physical paper; LyriSys [60] generates topically relevant song lyrics based on a syllabic pattern; Play Write [61] supports writing microtasks; StoryAssembler [62] supports writing dynamic / non-linear stories. Because the writing goals are quite diverse, these systems use a variety of technologies. Some are about providing text to the writer but most provide support in some other way, like aiding in structuring tasks.

As in planning and reviewing, the translating tools for highly constrained goals are more highly structured. Likely this structure is what allows the tool to be supportive, or is developed by designers to provide traction for the problem. We also saw these tools being quite niche. More general writing tasks like storytelling (e.g. MiL (stories) [53], BunCho [54], and Writing with RNN [63]) were lightly constrained, but this isn't inherent to storytelling. Subtasks within storytelling can be quite constrained, but we didn't see them turn up in our literature review. An interesting example of highly constrained translation that we didn't see is taking bullet points and turning them into prose. This is another example of a highly constrained but more general purpose task we believe is an interesting area for future work.

The tools studied had various levels of technical complexity and difficulty. They draw on a wide spectrum of user interactions and language technologies. They ranged from full document editors such as Microsoft Word and OmniFocus, which provide rich interface's on top of feedback such as spell checking, to collaboration software such as GitHub, to text generation technologies such as context-free grammars and neural algorithms. Figure 3.3 shows the distribution of tools according to complexity and level of constraint. For annotating the complexity of a tool we followed [47], where high complexity refers to an entire system or suite of tools, and low complexity refers

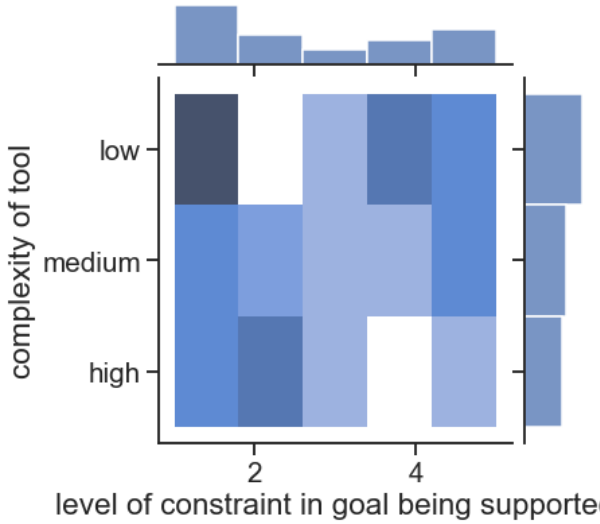


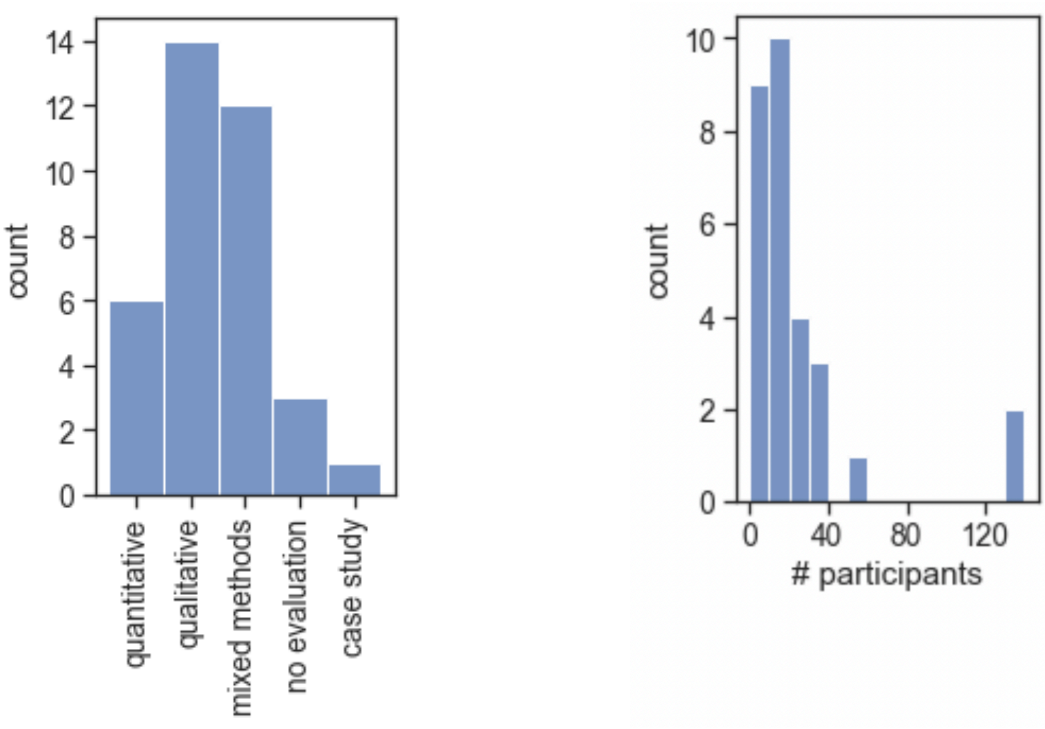
Figure 3.3: There were more tools with 1-2 features (low complexity). The distribution of constraints being supported was U-shaped.

to tools with only one or two features. (That is, complexity here is not a measure of technical difficulty.) The tools reviewed were slightly skewed towards low complexity (14 of the 33 tools). Most of the tools (78%) were contributions of the authors.

A third (11 of 33) of the tools used a neural algorithm for text generation or translation and five used some other form of grammar, template, or external knowledge source for text generation. BunCho [54] was one of the handful of non-English tools (5 of 33), using GPT-2 to generate Japanese story titles and summaries. Predictive text completion was used by a number of tools, like Storytelling Assistance [45], to insert text in a way that might provoke the writer to explore new directions and see their work in a new light.

Analysis of evaluation methodologies. A total of 33 evaluations were conducted among the 30 papers we studied. Several papers conducted more than one evaluation for their research, while three papers had no evaluation: Shakespeare [64], Dakje [58], and Ambient Letter [59].

Figure 3.4 shows the distributions of evaluation type and number of participants. On average, 25 participants were recruited for evaluation of writing tasks. 75% of the evaluations were conducted with fewer than 40 participants and these evaluations were either qualitative or mixed methods, likely because qualitative evaluations produce large and unorganized data that does not



(a) Evaluation Type

(b) Number of Participants

Figure 3.4: Histograms representing the distribution of evaluation methodologies.

allow easy manipulation and analysis for too many participants. Writing Together [49] and Storytelling Assistance [45] conducted studies with about 130 participants, and both were quantitative only evaluations.

Looking at the papers that had some component of qualitative evaluation, there was a wide range of criteria studied, including quality of writing, usability, usefulness, coherence to context, enjoyment, satisfaction, impact on flow, impact on confidence, and many more. Qualitative studies tended to assess their tools through semi-structured interviews with a small group of target users, such as creative writers or students. Around 50% of qualitative evaluations were done alongside a quantitative evaluation. Studies with only quantitative evaluations, such as Storytelling Assistance [45], assessed quality of the tool with questionnaires reported on a Likert scale and used measures specific to the tools they are studying, like Levenshtein edit distance or simultaneous time spent on writing, to evaluate user's attitudes and collaborative usage of the tool.

Around half of the evaluations reported did not include the time participants spent writing with

the system, which makes it difficult to assess this in relation to other aspects of the studies. Among the evaluations that reported time spent writing, quantitative evaluations done without the addition of a qualitative evaluation have a much shorter average time spent with the user (5-10 mins) than the others (25 mins). However, there's nothing inherent about quantitative or larger-scale evaluations that precludes writing for a longer period of time.

Quality of writing corresponds to a variety of different task-specific measures. MiL (stories) [53] has Amazon Mechanical Turk workers rate outputs for creativity, coherence, grammaticality, and entertainment. AL has annotators rate an argument according to a formal schema. Writing Together [49] studied writing done during a project writing course; writing quality was determined by course graders.

3.3.3 Evaluation Recommendations

Given so much variety in the evaluation methodologies, we make several recommendations on how evaluations could become more comparable:

- *Report more details of the actual writing* done in the study, for instance amount of time spent writing, amount of words written, and the type of participants recruited (novice, expert, etc.).
- *Use shared surveys* rather than develop new ones each time. The Creativity Support Index [65], NASA Task Load Index [66], and Technology Acceptance Model [67] may all be useful. We also encourage researchers to propose writing-specific surveys that can be used by others.
- *Report user interaction measures*, like edit distance, and number and frequency of interactions, that can be shared across different writing tasks.

3.3.4 Proposed Shared Tasks

Perhaps the biggest barrier for comparing research is the lack of shared tasks. These papers represent a broad range of writing tasks, from slogan writing to dynamic storytelling to argumentative writing. While we do not believe that writing is a monolith, and nor should be writing support

tools, a set of shared tasks may help consolidate the work.

We suggest three shared writing tasks: **story writing** (fiction), **argumentative essay writing** (nonfiction), and **personal essay writing** (creative nonfiction). Personal essay writing has many elements of fiction, like relying on character and narrative, while being constrained to the reality of the writer's lived experience. These tasks span from being completely open-ended (story writing) to partially constrained (personal essay) to quite constrained (argumentative essays). Within each task are many subtasks which span from being very open-ended (how to start the argumentative essay) to very constrained (how to describe an existing character).

We choose these tasks because they each contain goals which could span the entire design space and a variety of genres. There are many tasks we did not include, like emails, explainers, and poetry. These were not chosen because we felt they were too niche (like poetry) or too broad-reaching (like emails) to help unify research.

Below we discuss some variation within each task, and some potential subtasks to focus on:

Story writing. This already-common task contains within it diverse goals from plot development to scene description. The length can vary its complexity and they can be constrained to varying degrees by a prompt. We recommend two specific tasks. The first is writing stories in response to a prompt. (Again, this is already common and can be continued to be worked on.) The second is adding detail to an existing or partially written story, for instance adding character or scene descriptions. This will allow work to look at some of the more constrained parts of story writing.

Argumentative essay writing. This task is common in U.S. secondary education and can be extended to include journalistic forms like opinion pieces. It contains subtasks like defending propositions, writing an engaging introduction, and appealing to the audience. We recommend two specific avenues of research: supporting argumentative structure, and supporting introductory remarks. While supporting structure gets to complicated technical elements of the ideas of a piece of writing, supporting introductory remarks requires more modeling of the reader and understanding what makes text interesting and engaging.

Personal essay writing. This task can include private journaling as well as more public forms like memoir or even personal statements. It can contain subtasks like finding relevant historical information or identifying potential narratives. The utility of this task is how writers are self-motivated. For this task we recommend focusing less on the quality of writing, and more on the experience of the writer. While stories and argumentative essays have many formal elements that can be used in evaluation, we recommend this task be about immersion and self-expression.

3.3.5 Limitations

Our systematic review was limited in scope, as we focused only on the last five years, and our query for selecting papers may not have caught all relevant papers. For instance, one clear problem with using the ACM Digital Library is that many NLP conferences are not included. Future work should investigate more sources for papers, and look further into the archive. Additionally, we did not include commercial or open source writing tools that exist outside of the academy, which likely would improve the findings of any large-scale, systematic review of writing support tools.

There are also many more questions that could be asked about writing support tools. For instance, we found that user type was not widely reported, but user type may be implied by the writing task, or inferred by the evaluation methodology. Relatedly, further analysis could be done on how much work is dedicated to fiction v. nonfiction or short v. longer writing. We hope that by making our selected papers easily accessible, others may use this to do their own investigations with other focuses.

3.4 Conclusion

This chapter presents a design space for writing support tools based on a cognitive process model of writing. It reports on a systematic literature review, reviewing 30 papers from the last five years (2017-2021). We find that highly constrained planning and reviewing are under-studied areas, and we more formally motivate this thesis's focus on constrained, creative tasks as an under-explored and unsolved problem in the writing support tools space. We see that evaluation method-

ologies vary widely, and propose validated surveys and interaction measures as ways to make evaluations more comparable across systems. We also propose three shared tasks—storytelling, argumentative writing, and personal essays—to aid in propelling work on writing support tools forward.

Chapter 4: Metaphoria: An Algorithmic Companion for Metaphor Creation

This chapter reports on the design and evaluation of a system to support writers with writing metaphors about abstract concepts.

Based on a literature review, **coherence to context** is the biggest barrier to use for creative writing support tools [2, 68, 37]. Writers come to systems with ideas and intentions in mind, and often use systems part-way through writing, when context has already been established. However many systems, upon evaluation with user testing, often fail to adhere to this context, whether it be text already on the page or ideas writers would like to explore. A system that is coherent to context should provides suggestion that make sense given the task at hand.

Secondarily, writers do not want tools that make their writing sound the same as others [69]. Thus, suggestions that result in **divergent outcomes** for writers is crucial. A system that encourages divergent outcomes provides many compelling options and increases the variation in writers' work rather than propel all writers toward similar metaphors.

These goals map to previous methodology in HCI for the evaluation of generative drawing tools. Jacobs et. al. [70] evaluate their drawing tool on compatibility (coherence to context) and expressiveness (ability to express a divergent set of ideas) .

This work addresses these goals by designing a system to suggest metaphors about a topic a writer wants to write about. Inherent in this system is the idea that the writer provides a topic they are interested in, allowing the writer to set an intention; the system then need not uncover this intention but rather only respond to the intention coherently. More specifically, to address **coherence to context**, we focus on generating metaphorical connections for a given "seed metaphor". Seed metaphors are of the form *[source] is [vehicle]*, e.g. *envy is a bell*, where *envy* is the source (the thing you are trying to describe) and *bell* is the vehicle (the thing used for the describing). By focusing on connections between the words, such as 'envy can sound the alarm like a bell',

Figure 4.1: A poet using Metaphoria to find metaphorical connections between *america* and *wood*.

rather than the selection of the seed words, we leave open the possibility that the writer can input one or both words of the seed metaphor. To address **divergent outcomes**, the system generates and present multiple, distinct suggestions for each seed metaphor. This approach allows writers to select a suggestion salient for them in particular.

4.1 Related Work: Metaphor Generation

Metaphor generation is a version of conceptual blending [71] that has been correlated with general fluid intelligence [72] and is considered an important challenge in artificial intelligence research [73].

Current metaphor generation systems find properties that can be attributed to the two concepts in the metaphor. Two prominent algorithms are Thesaurus Rex [74, 75] and Intersecting Word Vectors [76]. Thesaurus Rex [74, 75] is a web service that provides shared attributes and categories for input concepts. For example, inputting *coffee & cola* produces results such as *acidic food* and *nonalcoholic beverage*. Thesaurus Rex is explicitly designed to support metaphor generation [77, 78]. Intersecting Word Vectors [76] is a metaphor generation algorithm in which connector words

are found using word embeddings. Connector words are those found in the intersection of the 1000 words closest to each of the concept words. For example, connector words for *storm & surrender* include *barrage* and *onslaught*. These systems are strong baselines for metaphor generation from the artificial intelligence and natural language processing communities.

Theories of metaphor often conform to structural alignment theory [79] in which analogies are discovered by finding isomorphic sections of knowledge graphs, where each edge is a structural relation between concepts. Work on using analogies for product design [80] has focused on the difference between structural and functional aspects of products for ideation. We draw on these ideas of structural and functional connections as a search function for concept attributes.

4.2 System Design

4.2.1 Generating Coherent Connections

Starting with a seed metaphor, our approach is to first generate many features of the vehicle (*bell*), and then rank these features by how related they are to the source (*envy*). This aligns with traditional metaphor usage, in which features of the vehicle are used to explain the source.

To find features of the vehicle we use ConceptNet [81], an open-source knowledge graph, as a source of structural and functional properties of words. Structural properties are elements that define or compose an object. For example, a *bell* has a *clapper* and a *mouth*. In ConceptNet, we select for structural features by querying the "HasA" relations of the vehicle. Functional properties focus on an object's actions and purpose. For example, a *bell* can *make noise* and be used for *alerting*. In ConceptNet, we select for functional features by querying the "UsedFor" and "CapableOf" relations. Together, structural and functional properties provide a large set of potential connections from the vehicle to the source.

Not all features of the vehicle (*bell*) will metaphorically map to the source (*envy*). To find the most relevant ones, we rank how related the vehicle features (e.g. *used for getting attention*) are to the source (*envy*).[1] To rank suggestions we use GloVe word embeddings [82] trained on Wikipedia

[1]Models that rely on intersection, selecting only those features which by some measure match the source concept,

high	envy is used for getting attention like a bell
	envy is for alerting you to something like a bell
	...
	envy is used to toll like bell
low	envy is for playing music like a bell

Table 4.1: Examples of connections with high and low relevance for the seed *envy is a bell*.

2014 + Gigaword 5. Word embeddings are a common way to measure the semantic similarity between words [83]. Here, we use them to measure the semantic similarity between the vehicle property and source word. Examples of vehicle properties with high and low relevance are found in Table 4.1.

To find the semantic distance between vehicle features and the source word, we use a modified Word Mover's Distance (WMD) [84]. WMD is an algorithm for finding the smallest distance between two documents, i.e. sets of words, in a word embedding space. It formulates distance between documents as a transportation problem: we denote $c(i, j)$ as the distance between words x_i and x_j, where $c(i, j)$ is the cosine distance between the two word vectors. Given two documents D_1 and D_2, we allow each word i in D_1 to be transformed into any word in D_2 in total or in parts. We denote T_{ij} as how much of word i in D_1 is transformed to word j in D_2; therefore $\sum_{i,j} T_{ij} = 1$.

We can define the distance between two documents as the minimum cumulative cost of moving all words in D_1 to all words in D_2. This is equivalent to solving the linear program

$$\min \sum_{i,j} T_{ij} * c(i, j) \tag{4.1}$$

for which specialized solvers have been developed. For example, this would find the shortest distance from *making noise* to *envy*.[2] From this ranking of connections, we can select the top n as the most coherent.

often do not produce any results. Given that our goal is to generate suggestions for a human writer, we are not as concerned about false positives and would rather return poor results than no results.

[2]In this usage, D_2 is always a single word, the source concept, although our implementation allows for natural expansion into multi-word sources.

4.2.2 Selecting Multiple, Distinct Connections

In order to promote diverse outcomes, our system presents writers with 10 coherent suggestions that are semantically distinct. For instance *get attention* and *getting people's attention* may both be coherent, yet they give effectively the same idea to the writer. For this reason, as we build our list of suggestions to show the writer, we throw out any feature that is too close to any of the features already ranked. This closeness is again calculated with the Word Mover's Distance, this time between two features. Through observation, we find a distance of less than 4 indicates two features are not semantically distinct.

4.2.3 Additional Coherence with Valence Ranking

Pilot testing showed that sometimes highly ranked features had a mismatched sentiment with the source concept. For instance, consider the seed metaphor *envy is a book*. *Envy* has a typically negative sentiment, while the feature *for learning from* has a typically positive sentiment. When this feature was highly ranked ('you can learn from envy like you can learn from like a book'), people found the mismatched sentiments to be jarring and caused them to lose faith in the system—even though upon further reflection a participant may appreciate that the experience of envy can be a useful learning experience. This issue of mismatched sentiments is caused by the fact that the word embedding space is not sensitive to antonyms; words with opposing sentiments (e.g. *bad* and *good*) can be close in the embedding space because they are nonetheless highly related. However, people who are first shown more intuitive features were more likely to appreciate the antonym features. Thus, we first select the suggestions as shown above, and then re-rank them by how similar the valence of each one is to the source concept.

Valence is the positive or negative connotation of a word and we assign valence scores to all words based on Warriner et al.'s database [85]. We denote the valence of the source as V_{source} and the valence of word i in the feature V_i for words $1, ..., n$. Then we define the valence distance as

$$V_{dist} = |V_{source} - \text{avg}(V_1, ..., V_n)| \tag{4.2}$$

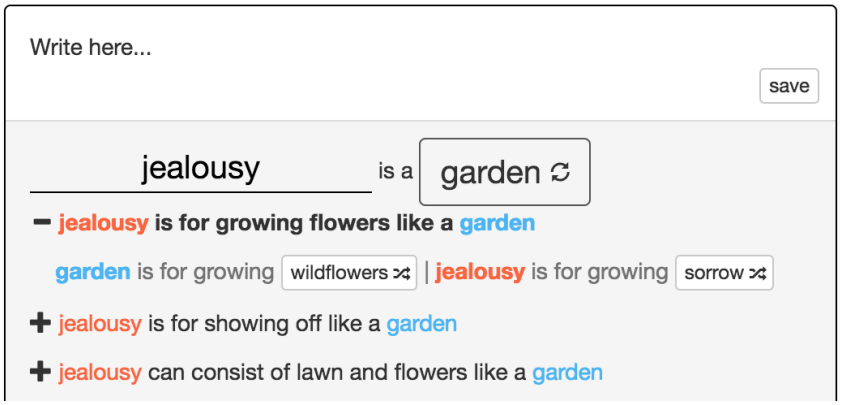


Figure 4.2: Screenshot of Metaphoria with suggestion for *jealousy is a garden* expanded.

We can then reorder the suggestions from the smallest valence distance to the largest.

Finally, we rephrase all connections into a suggestion for the writer; given the source *envy*, the vehicle *bell* and the connecting feature *making noise*, the suggestion is presented as 'envy is used for making noise like a bell'.

4.2.4 Additional Distinctness with Suggestion Expansion

Great metaphors are specific; we want to support writing specific metaphors by expanding them to include more details of how the source and vehicle are connected. If envy *makes noise* like a bell, we can expand on the details of the noise a bell makes (e.g. *vibrato, reverberation, high/low pitch*) and how these details relate to envy. For example, the noise of a bell has *reverberation*; and envy has *lasting bitterness*. Metaphoria provides multiple detailed metaphoric expansions for each suggestion to give writers more diverse options.

To generate the expanded metaphors, we first split each suggestion into two parallel sentences: one about the vehicle (*bells make noise*) and one about the source (*envy makes noise*). We want to find several alternative words to replace *noise* in each sentence. To generate these words, we again rely on word embeddings. This time, however, we want to discover words that will syntactically match the sentence—for this reason, we use word embeddings trained using a dependency parse as the context [86]. This results in similar words also having a similar part of speech. We use the word embeddings to create list of 60 words similar to the content word (*noise*) and 60 words

similar to source (*envy*). Then, we order these words by similarity to the vehicle (*bell*) and original word (*noise*), respectively, and return the 10 most related words in each case. Figure 4.2 shows the interface where a writer selects the suggestion "jealousy is for growing flowers like a garden" and can click through suggested expansions such as "jealousy is for growing sorrow."

4.2.5 Interactivity

The above methods are embedded in a Flask-based web application, as shown in Figure 4.1. Writers can input their own source and click through a set of common vehicles. Each combination will generate a list of up to 10 suggestions, and each suggestion can be expanded.

The design of Metaphoria has our goals of **coherence to context** and **divergent outcomes** in mind. By allowing writers to input a source and change the vehicle, we adapt to the intention of the writer, allowing greater coherence. Showing writers 10 semantically relevant suggestions, and enabling writers to 'shift' the suggestions with the detail words, enables a diversity of ideas and, hopefully, responses.

4.3 Study 1: Suggestion Quality

This study evaluates the quality of the suggestions Metaphoria generates. To achieve **coherence to context**, suggestions should make sense given their seed metaphor and enact principles of high quality writing.

4.3.1 Methodology

To evaluate the suggestions, we compare them to two other state-of-the-art metaphor generation algorithms: Thesaurus Rex [75] and Intersecting Word Vectors [76]. These algorithms are described fully in Section 4.1. As our system produces a ranked set of suggestions, we also compare both the highest ranked suggestions with the lowest to evaluate the effectiveness of the ranking algorithm.

Thesaurus Rex produces shared attributes; for example *envy & bell* produces attributes such as

Metaphoria
envy is used for getting attention like a bell
envy is for alerting you to something like a bell
Thesaurus Rex
envy is loud like a bell
envy is audible like a bell
Intersecting
envy is shiny like a bell
envy can behold like a bell

Table 4.2: Examples of metaphors from Metaphoria and two comparable, state-of-the-art metaphor generation algorithms for the seed *envy is a bell*.

loud. Intersecting similarly produces connector words; for *envy & bell* it produces words such as *behold*. In both cases we formulate these into sentences comparable with Metaphoria suggestions. Table 4.2 shows examples of this.

For each system we select the top three ranked suggestions. Ranking for Metaphoria is done using the WMD distance to the source concept (as explained in Section 4.2); both Thesaurus Rex and Intersecting generate ranked lists.

To compare the systems, we define three metrics for evaluating metaphor strength. The first is **aptness**, in which a metaphor accurately describes a connection between the concepts; this is the ground level of metaphors. The second is **specificity**, in which a metaphor describes a connection unlikely to be transferable other concepts. The third is **imageability**, in which a metaphor describes a connection the reader can visualize.

We expect that Intersecting will not be particularly apt as it relies solely on the embedding space to provide meaning and embedding spaces notoriously lack consistent discrete semantics [87]. Thesaurus Rex uses textual evidence, so we expect its connections to be apt, but because of this we also expect it to be less imageable and specific as it may only find higher level, and thus vaguer, attributes.

We have three hypotheses:

– H1: Metaphoria is **more apt** than Intersecting and **at least as apt** as Thesaurus Rex.

	Apt: makes sense
strong example	Love can *come on unexpectedly*.
weak example	Love is a *weather event*.
	Specific: uniquely belonging
strong example	Love can *last through the night*.
weak example	Love is *dark*.
	Imageable: evokes visual
strong example	Love can *rain down on our heads*.
weak example	Love can *scare people*.

Table 4.3: Examples of strong and weak sentences for each of the metaphor evaluation metrics. All sentences are based on the seed metaphor *love is a storm*.

– H2: Metaphoria is **more specific** than Thesaurus Rex and Intersecting.

– H3: Metaphoria is **more imageable** than Thesaurus Rex and Intersecting.

Additionally, we want to know if top-ranked Metaphoria suggestions are more apt than bottom-ranked ones. For this, we compare the top three and bottom three ranked suggestions. Our hypothesis is:

– H4: Top-ranked Metaphoria suggestions are **more apt** than bottom ranked ones.

We hire two professional writers with an MFA in Creative Writing as annotators. We consider 12 different seed metaphors, e.g. *hope is a stream*, and for each generate the top 3 metaphor suggestions from each system. Additional we generate the bottom 3 metaphor suggestions for Metaphoria. This results in 144 suggestions total.

The annotators consider each metaphor suggestion and mark whether it is apt, specific, and imageable. They are told that all suggestions are generated by computers, but they are not told anything about how or the fact that they come from different systems. They are shown the suggestions for each seed metaphor in random order.

In addition to definitions of the metrics, annotators were also provided with examples of positive and negative cases for each category, as shown found in Table 4.3.

As in any evaluation of linguistic artifacts, it is not clear that there are precise or correct rankings for all of these attributes. Instead, there are general trends that most native English speakers

	Apt	Specific	Imageable
Metaphoria (M)	97%	**82%**	**100%**
Thesaurus Rex (TR)	**100%**	47%	**100%**
Intersecting (I)	49%	43%	53%

Table 4.4: While both Metaphoria and Thesaurus Rex generate apt and imageable metaphors, only Metaphoria consistently produces specific metaphors.

Hypothesis	diff	t-value	p-value
H1a M *more apt* than I	0.48	5.83	<0.001
H1b TR *more apt* than I	0.51	6.16	<0.001
H2a M *more specific* than TR	0.34	3.36	<0.05
H2b M *more specific* than I	0.38	3.55	<0.001
H3a M *more imageable* than TR	0.00	n/a	n/a
H3b M *more imageable* than I	0.47	5.59	<0.001

Table 4.5: T-tests confirm that Metaphoria is as good or better across all metrics than state-of-the-art metaphor generation algorithms. P-values are Bonferonni corrected.

would agree with. We first have the annotators evaluate suggestions for 2 seed metaphors together and discuss their evaluation in order to establish common understandings of the metrics. They then individually evaluate the suggestions for the 12 seed metaphors.

4.3.2 Results

We report the percent agreement between the two annotators for apt, specific, and imageable (and the Cohen's Kappa correlation coefficients) to be 85% (0.63), 83% (0.67) and 88% (0.64), respectively. Given the natural ambiguity of metaphors and creative writing, this is a high level of agreement.

The following results are determined by combining the evaluations of the two annotators; the higher evaluation is used in cases of disagreement. Table 4.4 shows the percent of times a given systems' suggestions was marked as apt, specific, or imageable. While Metaphoria and Thesaurus Rex metaphors are both consistently apt and imageable, Metaphoria outperforms all systems on specificity.

	Apt	Specific	Imageable
Top-ranked	**97%**	82%	**100%**
Bottom-ranked	78%	**85%**	89%

Table 4.6: Top-ranked metaphors perform significantly better than bottom-ranked metaphors on aptness and imageability; there is no significant difference for specificity.

To test H1-3, we perform paired t-tests (Bonferonni corrected) on the relevant pairs and disprove the null hypothesis for H1 and H2. However, it is clear that H3 does not hold as both Metaphoria and Thesaurus Rex were 100% imageable. The results of the statistical tests can be found in Table 4.5.

Surprisingly, Thesaurus Rex metaphors were as imageable as Metaphoria ones. In general the annotators found adjectives like *hard* more imageable than we expected. However, Metaphoria still outperforms other systems on specificity.

We also consider the difference between the top and bottom ranked Metaphoria suggestions; Table 4.1 shows examples. Table 4.6 shows the percent of times a given systems' suggestions was marked as apt, specific, or imageable. Top ranked suggestions are more apt than bottom ranked ones (t = 2.49, p-value = 0.01) which confirms H4. There is no significant difference for specificity (t = -0.30, p-value = 0.76). However, top ranked suggestions are slightly more imageable than bottom ranked suggestions (t = 2.09, p-value = 0.04). It could be that aptness makes it easier visualize the suggestion.

This shows that Metaphoria creates high quality metaphors and can provide strong suggestions to writers.

4.4 Study 2: Novice Users

This study evaluates the quality of the suggestions Metaphoria generates in the context of a specific writing task: writing extended metaphors. This allows us to test **coherence to context**, as well as if Metaphoria supports **divergent outcomes** when writers are given the same list of suggestions.

4.4.1 Methodology

We recruited 16 undergraduates: 8 female, 8 male, with an average age of 19.5 ($\sigma^2 = 1.2$). Each participant did a writing task and a semi-structured interview.

Each participant was asked write a metaphor that expresses a connection between an abstract concept and concrete object presented to them. They are given the following example for the seed *love is a stream*:

> Love is something that just drags me along. Like a stream it just takes me in whatever direction it is going.

We present each participants with six seed metaphors. The metaphors are generated by combining a random word from a set of poetic themes (e.g. *love*) with a random word from a set of concrete nouns (e.g. *stream*) [76]. Participants are asked to write about these seed metaphors one at a time—3 with Metaphoria and 3 without. All participants were given the same seed metaphors in the following order:

- *gratitude is a stream*
- *peace is a king*
- *jealousy is sand*
- *consciousness is a shadow*
- *loss is a wing*
- *friendship is snow*

To counterbalance the experiment, half the participants use Metaphoria with the first three metaphors, and half use it with the last three metaphors. Figure 4.3 shows how the interface is presented in each case.

After the participant completes the task, we conduct a semi-structured interview in which all participants are asked the same set of core questions, with follow-up questions asked as specific issues come up. During the interview, the participant or interviewer could use the interface to go back and look at what the participant wrote, or interact with the suggestions again.

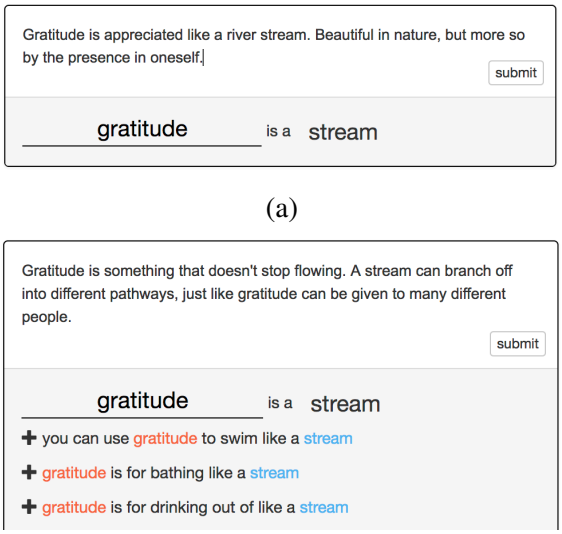


Figure 4.3: Interface for constrained writing task, in which participants wrote extended metaphors without suggestions (a) and with suggestions (b). Figure includes responses from P12 (a) and P10 (b).

In this study we are testing Metaphoria for coherence to context. If the suggestions are not coherent, participants will not be able to use them to write coherent sentences, which is their goal. Thus, usage is a strong signal for coherence. We also test for divergent outcomes by looking at the variety of responses. If Metaphoria does not support divergent outcomes, metaphors written across participants will be more similar when using Metaphoria than not.

4.4.2 Results

12 of 16 participants used the suggestions to the complete the task. Although all participants were given the same suggestions in the same order, they used a variety of different suggestions. For instance, given the seed metaphor *peace is a king*, P10 used the suggestion 'peace is for leading the people like a king' while P6 used the suggestion 'peace is for rallying the troops like a king'. Some participants were inspired by multiple suggestions, like P1 who used two suggestions, 'friendship is for beautiful vistas like snow' and 'friendship often arrives with a storm like snow', to write the following metaphor:

> Friendship often breaks out from kindness. It is a snow that often falls around christ-

mas.

Many participants were impressed by the quality of the suggestions, like P8 who said:

> "I like 'you can use gratitude to wash something like a stream'. That's something I wish I had come up with. That's creative."

Several of these participants acknowledged that the quality of the suggestions varied. P3 said that although some of the metaphors didn't make immediate sense, they thought that the metaphors could make immediate sense to someone else.

All participants were asked to choose one suggestion that was bad in some way and discuss why. Most participants spent some time rereading suggestions to select one. During this process, several participants discovered that a suggestion they previously thought did not make sense they could in fact interpret. P4 said:

> "With this one I was sort of a little confused, 'peace is for moving forward and backwards in checkers like a king', I guess it makes sense now that I say it out loud. It's saying that peace doesn't have any limits on it."

Of the 4 participants who did not use the suggestions, 3 said this was because the suggestions did not make sense. They often said the suggestions were too literal or simply nonsensical. However, P12 said the suggestions did make sense, but she did not want to use them because she wanted to demonstrate that she could write creative metaphors on her own. We come back to this in the Discussion section.

The suggestions may be coherent, but if participants end up writing very similar responses then Metaphoria is not supporting divergent outcomes for writers. We report both quantitative and qualitative results. To quantitatively measure this, we measure the variation of responses across all participants when they did or did not use Metaphoria. Here we define variation as the distribution of distances between all responses—high variation means all responses were very different from all other responses. We measure distance as the Word Mover's Distance between two responses.

	'gratitude is for bathing like a stream'
P6	Like a stream, you can bathe in gratitude and as the stream cleans your body, gratitude cleans your soul.
P13	A stream, to me, is rapid and powerful and has the ability to sweep you away. Gratitude offered by a friend or even a stranger is a stream in this way; it has the unexpected power to swell your heart with positive emotions and completely sweep you away.
	'jealousy can irritate skin like sand'
P16	Jealousy is a sand. It finds a way to irritate and conflict trouble of mind upon those whom it possesses.
P2	Jealousy can itch and irritate your mental behavior similar to the sand that clings on to your clothes and feet.

Table 4.7: Metaphoria mostly resulted in distinct responses, even when writers used the same suggestion, as in the 'gratitude' examples. But sometimes suggestions resulted in very similar responses, as in the 'jealousy' example.

The responses without Metaphoria act as a baseline for the variance we expect to see in the responses. If participants were staying close the suggestions, as opposed to expanding or shifting the ideas, we would expect there to be less variation with Metaphoria. Less variation means similar ideas, words, and phrasing. As a reminder, all participants received the same suggestions when they had access to Metaphoria.

Our hypothesis is as follows:

– H5: The variation in responses with Metaphoria is at least as large as the variation in responses without.

We compare the variation per seed metaphor with and without Metaphoria. There is no significant difference in the variation of the responses for 4 of the 6 seed metaphors. For *consciousness is a shadow* there is significantly greater variation with Metaphoria; for *jealousy is sand* there is significantly greater variation without. Table 4.7 shows examples from participants who said they were inspired by the same suggestion, demonstrating the wide range of directions participants took the idea, as well as examples of the more convergent responses.

Qualitatively participants did not feel like the suggestions boxed them in but rather inspired them to come up with new ideas. P4 expressed well how he would be inspired by a suggestion:

"I saw 'gratitude is for bathing like a stream' and that made me think, well, how big is

a stream? It started making me think about its size."

To demonstrate how far he took this idea, here is his final response to *gratitude is a stream*:

> Gratitude can be difficult to feel, or to notice, much like a stream that runs down the gutter of the road in a rainstorm. And like all streams, it can easily run dry–and you might not realize it's gone until it's too late.

We were worried that certain suggestions would be far more coherent than others, or that there would be a strong ordering effect, and therefore participants would always choose the same suggestions and write similar responses. However, as seen in the above analysis, this was not the case. Even when participants chose the same response, they would write radically different things.

4.5 Study 3: Expert Writers

This study evaluates if Metaphoria can adapt to a writer's own goals, and tests the system on inputs we did not expect. Our previous studies show Metaphoria is **coherent to context** and produces **divergent outcomes**; now we tackle whether these properties hold in real tasks which span a wide range of writer intentions.

4.5.1 Methodology

We gave three professional poets a 15 minute tutorial of Metaphoria and then asked them to write a poem on a subject of their own choosing using Metaphoria in some way. The poets wrote for around 30 minutes each. We then conducted a semi-structured interview, and utilized having Metaphoria available to discuss their process and response.

In this study, we gave participants access to the full interactivity of Metaphoria: they could enter in their own source concept, as well as a generate new vehicles, which are drawn randomly from a list of common poetic vehicles.

The poets were recruited through a mailing list for current and past MFA in Creative Writing students at a local university. All had a regular writing practice, were published poets, and one also held a teaching position in which they taught poetry writing workshops to undergraduates.

4.5.2 Results

Coherence to context All poets used several of the suggestions in their poem. Part of each poem is reproduced in Table 4.8, where words they input into Metaphoria are highlighted in pink and phrases from the suggestions they used are highlighted in green.

The context each poet brought to Metaphoria was very different. PO1 initially entered the word *island*; the first line of their poem was inspired by the suggestion 'island can fill a glass like wine', though they first spent several minutes with other suggestions like 'island can travel over water like a ship' and 'island can age over time like wine'. PO2 was initially inspired by suggestions for the seed metaphor *work is a garden*, where *work* was input during the tutorial; several words in the first stanza came from the suggestions for this seed. Later they input the words *swaying* and *she*.

PO3 brought a very different type of context. They input many more words than the other two poets, more interested in finding interesting suggestions than crafting a poem with a particular direction; almost every line derives from some part of a Metaphoria suggestion. They first input *sales*, then *marketing*, before exploring the word *metaphor*. Their first line is inspired by the suggestion 'metaphor is for restoring quiet like a bell'. Later they input words like *time*, *guns*, *history*, *elections*, *laughter*, and *stone*, to mention only a small number.

All poets found suggestions that resonated with them, though they were discriminate and often searched through several seeds before finding something they used. However, there were clearly different styles of use: PO1 and PO2 composed poems with some kind of linear narrative or thought, and used Metaphoria on words they had already written, often finding a suggestion that would finish the line they were working on. In contrast, PO3 input words they thought might be make for interesting metaphors, or words they simply overheard (we met in a coffee shop), many of which never made it into the poem. PO3's use was more like collecting interesting phrases,

PO1's response	PO2's response	PO3's response
My island fills glasses like wine, i'ts vines wrap around my new mouth like grapes. This new America is used to building things, anew, strange comfort like the rest of an air-bed at dusk. How new is new?	Garden Work with my mother, her tulips flaming blue and yellow, laboring to bloom beneath her palms, the soft lawn grating against early spring. We are wasting time, lingering under the porch light before dark, flirting with enemy weeds before my father returns home, drunk and swaying like a storm. She is used for currency and jewelry and lighting the pathway. She is for making flowers rise up to collide with her hands.	Metaphor for restoring quiet Use a gun to paint a room Addiction can clog a sink drain like hair History can win a war The garden of wasted time Fear to extinguish a fire like sand ice is for finding the source of light swimming is like snow. it is for children You can use caution to build fear in a movie You can use witchcraft to listen to music like an ear Corruption can outrun you like a horse

Table 4.8: Part of responses from three professional poets working with Metaphoria. Words highlighted in pink were input into Metaphoria by the poets, while words and phrases highlighted in green were suggestions that poets used.

which they then arranged and edited.

Divergent outcomes The resulting poems were of dramatically different styles, both due to each poet's differing usage of Metaphoria and their different writing styles. When explicitly asked about the expressiveness of the system, all poets noted that established writers have their own style and the system was unlikely to dramatically change it. Both PO2 and PO3 thought Metaphoria would increase the creativity of amateur poets, who tend to get stuck in cliche language; they thought the unexpectedness of the word combinations was likely to help.

However, PO2 did bring up concerns of ownership. While they did not think that Metaphoria limited them, they were concerned about using suggestions from Metaphoria that were too different from their intention, even if these suggestions were very good. PO3 used Metaphoria most liberally, yet had no such concerns. They drew a comparison between Metaphoria and Instagram, noting that while Instagram has produced a genre of photography that is very recognizable and thus the photos are somewhat similar, it has also produced unexpected and creative artworks. They speculated that Metaphoria might create a genre of Metaphoria-style poems, but would also allow poets to move in new and exciting directions. We analyze these concerns in the Discussion.

4.6 Discussion

4.6.1 Ownership concerns and cognitive models of usage

Ownership is extremely important to writers. It is essential that writers feel like they own their material, and Metaphoria was designed to augment writer's abilities, not replace them. To tackle this head on, we asked all participants about how much ownership they felt for what they wrote. Each poet in the expert study discussed their relationship to Metaphoria using a different cognitive model:

PO1 was unconcerned about the influence of the system on their writing; they thought of Metaphoria "like a calculator for words." They used Metaphoria as a **cognitive offloading tool**, outsourcing specific moments of word generation and allowing them to focus on other goals like

the overall direction of the poem and the flow of the lines.

PO2 was concerned about using Metaphoria when it produced particularly good images. For example, they thought the line 'she is used for currency and jewelry' was "an amazing line of poetry" but "definitely altered the direction of the poem," which worried them. In this case, they treated Metaphoria as a **co-creative partner** who contributed more to the poem than PO2 felt comfortable with.

PO3 used Metaphoria much more liberally–with no particular intended direction, they were more playful and wanted to uncover interesting Metaphoria-style combinations. In this case Metaphoria was used as a **casual creator** [88], an interactive system that encourages exploration in the creation or discovery of surprising new artifacts.

In the novice study, 4 of the 16 participants said that they felt less ownership over the final results because some amount of work was being done by the system; this reaction was strongest in those that thought the suggestions were particularly good. In this case, likely they saw Metaphoria as a **co-creative partner** contributing too much to their work.

Thus algorithmic suggestions are used differently depending on the cognitive model users project–a offloading tool that does grunt work (like a dictionary or thesaurus), a true partner that can do too much or too little, or a casual creator that allows the user to explore. Systems designers should be aware of different cognitive models and build tools that support creators without threatening their agency.

4.6.2 Design implications from ownership concerns

All participants in the novice and expert studies acknowledged that they happily accept prompts, ideas, feedback, and edits from people (both teachers and peers) without feeling loss of ownership. For machines to become acceptable co-creative partners, there are two design avenues:

Increased transparency can make the mechanisms of the machine more apparent. This way it feels more like a 'word calculator' than a system trying to outsmart you. Presentation of the suggestions may matter; more studies should be done on how this affects perceived ownership. It

could be that for some writers full sentences (even ones constructed naively from templates) are more threatening than a key dangling phrase.

Increased interactivity integrates the person into the creation process. The more interaction, the more the machine can be seen as a causal creator that helps explore new spaces. This interaction with a computational system can give people comfort and agency, similar to how we learn to converse with people offering us advice. Systems could draw suggestions from different contexts or genres that writer can pick or specify, such as a particular novel, technical text, or set of tweets, and include tunable parameters, such as suggestion length, vocabulary sophistication, connotative constraints (like negative/positive), or phonetic features.

4.6.3 Limitations and future work

Interaction with Metaphoria is limited to inputting a source word and requesting a new the vehicle word. This does not take into consideration what a writer has previously written, either the text of whatever they are currently working on or past work that might be relevant. To make systems more personalized, we could highlight how suggestions relate to a writer's previous work, or phrase suggestions in a syntactic style specific to the writer.

Additionally, Metaphoria can be expanded to other domains like journalism. For example, we can provide suggestions to metaphorically explain scientific concepts for lay people. "*CRISPR* can cut *genes* like *scissors* can cut *paper*." We can adapt the system by training a custom word embedding to provide representations for words in specialized domains, like medical research, technology, or law.

4.7 Conclusion

Motivated by past work on user-centric creativity support, this chapter presents Metaphoria, an interactive interface for generating metaphorical connections. Evaluations demonstrated that Metaphoria generates suggestions coherent to context and supports divergent outcomes for writers. We discuss ownership and cognitive models in human-computer collaboration, and present future

work for more interactive and transparent systems that can further empower creators.

Chapter 5: Sparks: Inspiration for Science Writing using Language Models

In this chapter I report on how language models can be applied to a real-world, high-impact writing task: science writing. This introduces challenges different to those in traditional creative writing tasks, such as my prior work with Metaphoria, which tend to deal with common objects and relations. Science writing requires a system to demonstrate proficiency within an area of expertise.

As a test-bed, I use a science writing form called "tweetorials" [**gero2021lightweight**, 89]. Tweetorials are short, technical explanations of around 500 words written on Twitter for a general audience; they have a low-barrier to entry and are gaining popularity as a science writing medium [90].

5.1 Related Work: Science Communication on Social Media

Science communication helps the public understand scientific contributions. It has been applied to vaccine misinformation [91], the COVID-19 pandemic [92], and climate change [93], to a name a few prominent instances. Traditionally, science communication took place through journals, conferences, articles, books, television and radio—places where peer review or editorial oversight was an implicit part of the publication process. However, the rise of digital networks and the ubiquity of social media presents opportunities for scientists to have direct channels to the public. Now any scientist can conduct science communication by posting about their work online [90], engaging in the 'Ask' communities on Reddit [94] or explaining a topic on YouTube [95].

This emerging trend, where the scientist can now partake in conversations outside of a gated process, reflects one of the many broad shifts away from traditional science communication. Scholars have reified this emerging form of communication as "post-normal science communication" [96]. Defining characteristics of post-normal science communication include a tolerance for sub-

jectivity, an insertion of the self, the integration of advocacy, and call to actions. Despite these dramatic shifts, the original tenets of science communication such as storytelling, analogies, figures, and citations remain valuable, and storytelling in particular is a driving principle within our system. Our work engages with post-normal science communication by exploring how new technologies might help people partake in online science writing.

5.2 Formative Study

In order to understand how a language model might best support the task of writing a tweetorial, I ran a formative study where participants were first given a technique for coming up with a compelling introduction, before being asked to write the first tweet of a tweetorial on a technical topic they were familiar with. Since the first tweet tends to set up the context and intention of the tweetorial [89] we expected this to be an effective and efficient way to understand what participants found difficult in the writing process, even when provided with writing strategies.[1]

5.2.1 Methodology

We recruited 10 students from our institution's Computer Science department (6 women / 4 men; 7 undergraduates [no first years] / 3 PhD students). Participants went through a tutorial on how to write an engaging introduction on two example topics—recursion and virtual private networks—which included several examples and a step-by-step process for coming up with ideas. The tutorial was developed in consultation with a science writing instructor and presented the following process for writing an engaging first tweet: 1) brainstorm three concrete situations related to the topic, 2) turn each situation into a question for the reader, 3) select the most engaging question and revise.[2] The tutorial was intended to provide the participants with as much "unintelligent" support as possible, mimicking what would be taught in a graduate-level science writing class,

[1] Initially we thought we could also run our final user study by asking participants to just write the first tweet, as we expected this to capture many of the creative aspects of tweetorial writing. However, a methodological finding was that writing the first tweet alone lacked some of the writing details we hoped to study as participants were not required to think through how they might actually continue from the structure they set up in the first tweet.

[2] The tutorials can be found at `http://language-play.com/tech-tweets/tutorials`.

such that we could identify where language models may be able to add benefit.

After the tutorial, participants were asked to select a topic from one of six Computer Science topics and write the first tweet for a tweetorial that would explain that topic.[3] Participants were asked to think aloud during the writing process and were not allowed to browse the web. Afterwards, they were asked a series of questions about their writing process in a semi-structured interview. The research team reviewed their writing with a science writing instructor. No formal coding was done, but general areas of success and areas for development were discussed.

5.2.2 Results

Although we never used the words 'creative' or 'creativity' when describing the task to participants, many participants reported that the task was difficult because it required creativity to come up with something that would engage the reader. Most participants said they don't typically do creative writing, so they found the task difficult and outside of their area of comfort. This supported our selection of tweetorials as a writing task, as we want to study a task that is both constrained and creative.

Participants found the tutorial helpful for a variety of reasons. Some liked seeing the examples, some appreciated a process to follow, and others found it comforting to see writing improve with brainstorming and revision. Several commented that the tutorial made the task look easy, but when they wrote about their own topic it was surprisingly difficult. 9 out of 10 participants said that making the topic interesting to a general audience was the most difficult part of the writing task. When pressed to be more specific, participants mentioned coming up with concrete examples/situations and creating an engaging question as hard tasks. Though this was influenced by the process the tutorial introduced, this confirmed that tutorials are not enough to fully support writers in this task.

When reviewing what the participants had written, all the tweets mimicked the tone of the examples. However, the science writing instructor had critiques for all of them, and most of the

[3]The topics were: hashing, sorting algorithms, Bayes theorem, HTTP, transistors, and Turning Machines. We selected these topics as ones that a) most computer science students should have learned in a formal setting, and b) could reasonably make for an interesting tweetorial.

critiques at the core were the same: the tweet lacked suspense. By this he meant, the tweet did not introduce a compelling problem or gap in the reader's understanding that would make the reader want to read more. Often this was because the example used wasn't particularly compelling or didn't reflect a real use case of the topic. Additionally, participants tended to repeat similar ideas to others who had selected the same topic.[4] Given that participants reported coming up with ideas difficult, it's likely that participants could have done better if given help with brainstorming.

We also noted that many of the tweets might be difficult to turn into tweetorials. For instance, some tweets engaged the reader with a question, but answering this question wouldn't require an explanation about the chosen topic. For this reason, in future studies we had participants write more than just the first tweet.

5.2.3 Design Goals

Based on our formative study, we developed two design goals:

1. **Support writers with idea generation.** Given that language models have no model of truth, we want our system to come up with "sparks", intended to spark ideas in the writer, rather than having the system provide the ideas themselves. This aligns with prior work on creativity support tools, where users make use of system outputs as initial directions that are then interpreted and diverged from in the users' actual creation [97].[5]
2. **Generate outputs that are coherent and diverse.** In order for writers to make use of outputs, even if they are not always perfectly accurate, they should be coherent: well-formed and generally reflecting accurate knowledge. Additionally, to support idea generation, outputs should also be diverse, such that writers have a variety of outputs to make use of.

[4] e.g. all the participants writing about HTTP used either Google or Twitter as their example, suggesting that people may converge on similar, easy to reach ideas.

[5] Additionally, this encourages the writer to feel more ownership over their final product, which has shown to be a concern in past work [98].

5.3 System Design

5.3.1 Generating Sparks

Language Model Selection To generate sparks we use GPT-2, an open source, mid-sized (1.5 billion parameters), transformer language model trained on 40GB of text from the web [14]. We use the huggingface implementation [99]. While larger open source models are available (though only to some),[6] we wanted to limit the size of the model we used as larger models are more expensive to run and take more time to generate text. Additionally, there have been many critiques of the super-large language models [19], and thus we wanted to use the smallest language model able to perform well for our use case. Anecdotally, we found that DistilGPT2, a 'distilled', smaller version of GPT-2 [100], was not able to produce coherent responses to our prompts. We experimented with fine-tuning GPT-2 on a small dataset of science writing, but found that this made little difference, especially compared to modifying the decoding method or the prompts. For this reason most of our design effort focused on decoding and prompt engineering.

Decoding Method In addition to selecting a model, we had to design a decoding method—how to select the next token given the probability distribution the model outputs. There are several common ways of decoding from language models: greedy search, beam search, top-k sampling [17], and top-n sampling [21], to name a few. Different methods have different strengths and weaknesses. Greedy search, which selects the most likely word at each generation step, is rarely used for creative text generation as it tends to produce very generic responses (and rarely finds the most likely sequence of words). In contrast, beam search, which maintains a 'beam' of n possible outputs, can find more likely sequences and tends to produce high quality results [22]. When trying to generate multiple possible outputs for the same prompt, sampling methods, where words are sampled from the language model according to their likelihood, are often used. However this often decreases the coherence of the outputs, because very unlikely words can now be generated

[6] For example, at the time this work was done, GPT-3 [15] was only accessible to those that had been granted access by OpenAI.

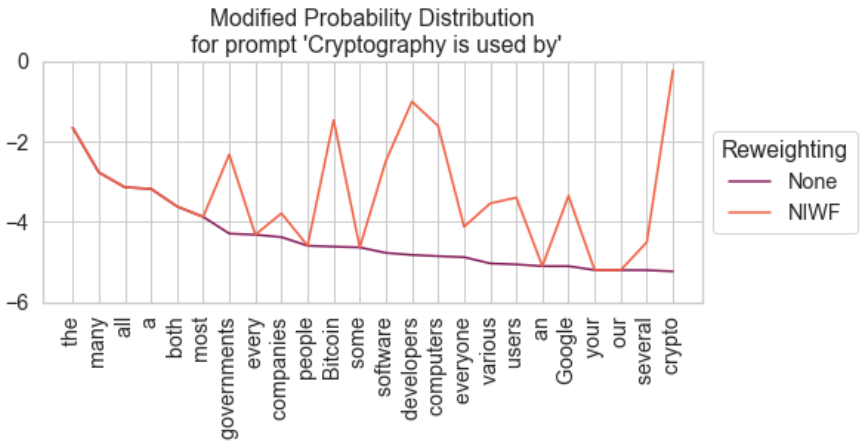


Figure 5.1: This graph shows how the likelihood of the 20 next most likely words given the prompt "cryptography is used by". The purple line shows the original probability distribution. The orange line shows the distribution after it has been rewieghted with normalized inverse word frequencies (NIWF). Words like “governments”, “Bitcoin”, “software”, and “developers” have an increased probability, while words like “many”, “both”, and “all” are not modified.

with some (albeit small) probability. For the purposes of having multiple unique sparks for our task, we designed a method that attempts to further increase the coherence of beam search while also increasing its ability to generate diverse outputs.

First, we modify the probability distribution using a normalized inverse word frequency, in order to increase the likelihood of infrequent words. Normalized inverse word frequency is often used in natural language generation to improve the specificity of outputs [101, 102], which is one method for increasing the overall quality of results. Here, we use normalized inverse word frequency purely during decoding as opposed to during training [103]. To calculate the word frequencies, we wanted a corpus that doesn’t over-represent uncommon science words, like a science writing dataset might, but also reflects modern word usage. For these reasons, we use a corpus of Vox news articles that includes all articles published before March 2017.[7] Figure 5.1 shows an example of the probability distribution being modified. In this figure you can see that words like “governments”, “Bitcoin”, and “software” have increased weight, while words like “many”, “both”, and “all”, are not modified.

Second, we use only the top 50 highest ranking tokens. This is sometimes called top-k sam-

[7] https://data.world/elenadata/vox-articles

pling, as only the top k tokens are used [21]. However, since we're not using a sampling method, the effect of this is to ensure that the modified probability distribution doesn't introduce any incoherencies by dramatically increasing the rank of a token very far down in the original probability distributions. For example, Figure 5.1 shows that the probability of tokens related to 'cryptography' are dramatically increased; if this occurred when the token 'crypto' was ranked, say, 200th in the probability distribution, it may introduce incoherencies.

Third, we increase the diversity of outputs by forcing the first token of each output to be unique, but attempt to retain coherence by generating the rest of the tokens with beam search. While several more sophisticated methods have been proposed to increase diversity while retaining the coherence of beam search (e.g. [104]), in testing we found none were as effective as simply enforcing the first token to be unique.

Finally, in order to keep the sparks succinct and generating quickly, we only generate 10 tokens after the prompt and cut off the generation as soon as a sentence has been completed. We implement our decoding method using the huggingface transformers [99].[8] Step-by-step enumeration of the decoding process, and further development details, can be found in the Appendix.

Prompt Engineering We craft a 'prefix' prompt to pre-pend to any prompt used by a writer. Prefix prompts have been shown to greatly improve performance by providing the language model with appropriate context [23]. We found early on in development that simply providing the model with a technical topic was not enough—also providing a context area was necessary for it to appropriately interpret technical terms. For instance, if you use a prompt like "Natural language generation is used for", the model is likely to talk about linguistic research on languages, rather than computational methods. If instead you use the prompt, "Natural language generation, a topic in computer science, is used by" the results are much more likely to refer to computational language generation. Given this, we pre-pend all prompts with the following: "{topic} is an important topic in {context area}" where {topic} and {context area} are provided by the writer.

In hand-crafting our prompts, we wanted to make sure our prompts captured a range of relevant

[8]The repository with the code can be found at `https://github.com/kgero/tech-tweets`.

Table 5.1: Prompt templates for science writing task.

category	prompt
expository	One attribute of {topic} is Specifically, {topic} has qualities such as
instantiation	One application of {topic} in the real world is {topic} occurs in the real world when
goal	For instance, people use {topic} to {topic} is used for
causal	{topic} happen because For example, {topic} causes
role	{topic} is used by {topic} is studied by

angles, so our system could flexibly work with any technical discipline. To do so, we synthesized work from expository and narrative theory into prompts capturing five categories: *expository, instantiation, goal, causal,* and *role*. Each category represented an angle that a writer might want to explore. All prompts can be seen in Table 5.1.

We manually developed these prompts according to established frameworks within narrative and expository theory. Our prompts within the categories of *instantiation, goal, antecedent,* and *role* draw upon the constructionist framework of inferences [105], specifically the following categories: case structure role assignment, causal antecedent, the presence of superordinate goals, and the instantiation of a noun category (respectively). Less formally, *instantiation* prompt templates suggest completions that instantiate where and in what ways topic X may occur in the real world. *Goals* prompt templates suggest completions that represent how topic X is used in the real world. *Causes* prompt templates suggest completions for how topic X might interact in cause and effect chains. *Roles* prompt templates cover entities involved with topic X. As tweetorials exhibit both elements of narrative and expository writing, we also borrowed signal phrases from Meyer's framework for expository text [106]—e.g. “specifically", “such as", “attribute”—and folded them within our prompt templates.

In testing we found that participants often wanted to follow up on an output by entering in their

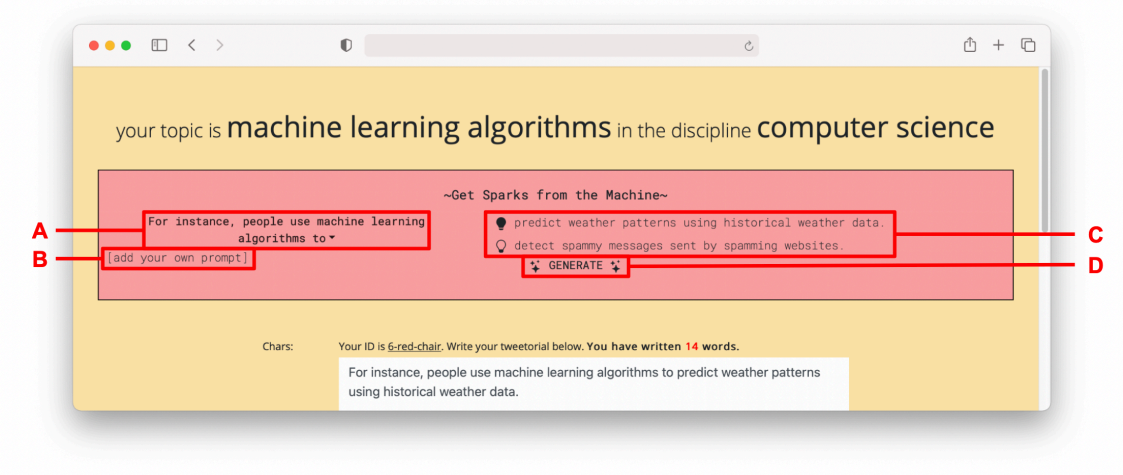


Figure 5.2: Example screenshot of our system that generates sparks. A: writers can select from 10 templates of prompts in a drop-down menu. B: writers can add their own prompt to the drop-down menu. C: sparks are generated with a lightbulb icon to the left; if writers click the lightbulb it will highlight and the spark is copied into the text area. D: writers can press the generate button in order to generate a new spark.

own prompt. For this reason, we added the ability for writers to add their own prompts, though this prompt would also be pre-pended with our prefix.[9]

5.3.2 Interface

Figure 5.2 shows a screenshot of the system with its important features marked. The website consists of a single textbox for writing, and a 'prompt box' above it that allows writers to interact with the sparks. Writers can select a templated prompt from a dropdown menu, or type in their own prompt and add it to the dropdown list. When a prompt is selected, if they press 'GENERATE' the language model will generate a single spark. Writers can 'star' a spark by clicking on the lightbulb icon—this fills in the lightbulb and also pastes the spark into the textbox. If a writer selects a different prompt, the sparks already generated are preserved such that if they return to a previous prompt their generated sparks will be shown again.

The writing area textbox contains some features useful for the tweetorial writing task. The

[9]One intriguing area of research is 'meta-prompting' [23] or 'chaining' [107], where the language model is used to generate the prefix for the next generation. While we found that this produced intriguing results for our use case, for example by having the model first produce a list of types of people who interact with a topic, and then putting those phrases into a downstream template, we thought it added too much complexity.

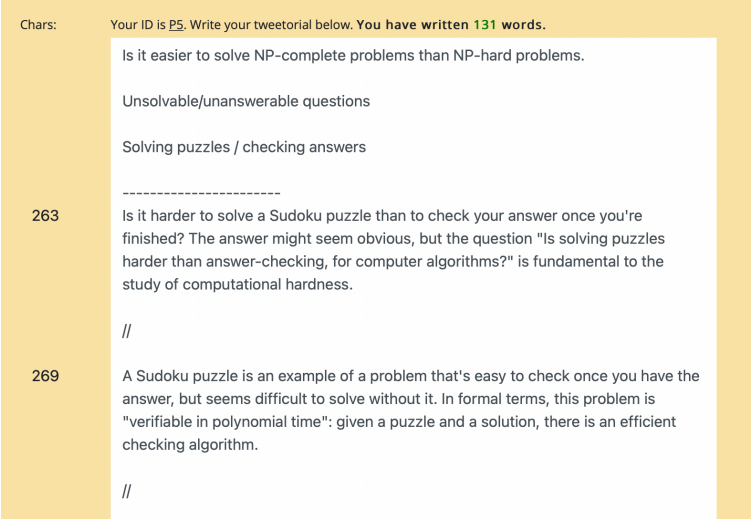


Figure 5.3: Screenshot of the text area from our user study. At the top is a word count, which counts only the words below the dashed line. Text above the dashed line is interpreted as brainstorming or notes. Participants can separate tweets with a double '//', and the character count for each tweet is shown to its left.

textbox is split into two sections with a line of dashes. Above the line is reserved for brainstorming and notes, a feature writers requested and found useful during pilot studies. Below the line is the text area for the tweetorial writing. A word count for the writer's tweetorial draft is displayed at the top of the textbox, and a character count for each tweet (separated by line breaks and two forward slashes) is displayed to the left. Figure 5.3 shows these features with an example from our user study.

The website is implemented using Python 3.7 and the Flask web framework.[10]

5.4 Study 1: Spark Quality

We wanted to evaluate how well the sparks in isolation (i.e. not in a writing task) met our design goals of generating coherent and diverse sparks. We also wanted to test how well the sparks could support a wide range of topics, and if certain prompts supported some topics better than others. To do so, we compared the sparks generated by the custom decoding method to a baseline system, as well as a human-written gold standard.

[10]A demo can be found at `http://language-play.com/tech-tweets/enter-topic`

We have three hypotheses:

– H1: The custom decoding produces more coherent and diverse outputs than a baseline system, but less coherent and diverse outputs than a human-written gold standard.
– H2: The custom decoding performs consistently across many different topics.
– H3: There is significant variance across output quality in topic+prompt combinations.

5.4.1 Methodology

We wanted to evaluate the quality of ideas for a variety of topics. We selected three disciplines that have a glossary of terms page on Wikipedia, and that have been demonstrated to be a rich discipline for science writing on social media.[11] These disciplines were computer science, environmental science, and biology. For each discipline we randomly sampled 10 topics from their glossary of terms page. See the Appendix for the full list of topics studied.

Collecting a Human-Written Gold Standard We wanted to collect human responses to our prompts to represent a gold standard or upper limit on the quality of ideas these prompts can generate. To do this, we recruited 2-3 PhD or senior undergraduate students in each discipline and had them complete the same prompts the language model did. These students acted like 'perfect' language models, with access to relevant expertise and a human-level understanding of how to write high quality sentences. Each student was paid $20/hour for as long as it took them to finish the task.

We explained to them that the purpose of the prompts was to generate ideas to support an expert writing about the topic for a general audience. Each student had to complete 5 prompts per topic in 3 different ways and was told to make the completions for a given prompt+topic combination maximally different (to encourage diversity). They were also instructed to ensure their completions were accurate, given their understanding of the topic, and that they could reference the web if they needed to check anything, as well as use web search results for inspiration. Finally, we explained that their ideas should be as concrete and specific as possible. Each student completed 5 prompts

[11] e.g. https://twitter.com/dannydiekroeger/status/1281100866871648256, https://twitter.com/GeneticJen/status/897153589193441281, and https://twitter.com/meehancrist/status/1197527975379505152

for the 10 topics in their discipline, for a total of 5 x 10 x 3 = 150 completions per person. It took them on average 3.5 hours to come up with completions for all 10 topics in their discipline, and in the end we had 6 high quality completions per prompt+topic combination.

Baseline Language Model Condition We compare the custom decoding to a language model baseline: group beam search with hamming diversity penalty. This is a strong baseline that encourages diversity in the way Vijayakumar et al. [104] recommend, and can be implemented using arguments in the 'generate' function in the huggingface transformer library. Both the custom decoding and baseline model use the same underlying language model.

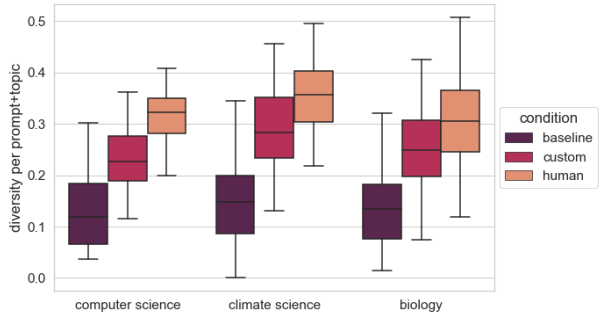
Measuring Coherence and Diversity Coherence is notoriously difficult to measure automatically; measures like perplexity measure an output's likelihood under the model itself. For this reason we recruited domain experts to annotate outputs for coherence on a 0 - 4 scale, in line with knowledge graph evaluations [108]: 0 ("Doesn't make sense"), 1 ("Not true"), 2 ("Opinion/Don't know"), 3 ("Sometimes true"), and 4 ("Generally true").[12] For biology, we had 3 senior undergraduate students majoring in biology; for environmental science, we had 2 senior undergraduate students majoring in environmental science; for computer science, we had 2 PhD students from the computer science department.[13] Each discipline had 900 sentences to annotate (300 human generated, 300 from the baseline model, and 300 from the custom decoding). 250 randomly selected outputs from each discipline were annotated by two different annotators, and the Cohen's weighted kappa was calculated as: $\kappa = .54$ for biology, $\kappa = .51$ for environmental science, and $\kappa = .46$ for computer science. Given that the agreement was moderate, we had a single annotation for the remaining sentences.

We also want to measure diversity, that is, for a set of outputs for a given prompt, how different are they from each other? Redundant or too similar outputs do not contribute new ideas to writers. We measure diversity with sentence embeddings specifically designed to elicit semantically mean-

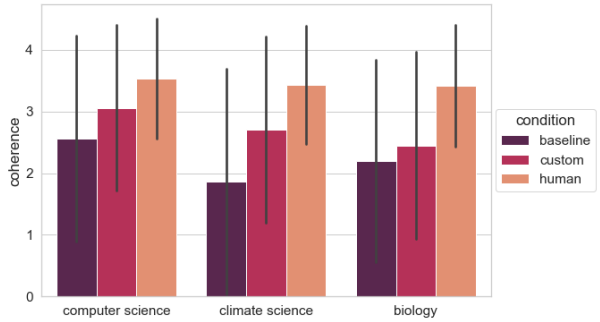
[12]This measures both coherence and cohesion, to lessen the load on annotators.
[13]The students could not have also participated in the generation portion.

Table 5.2: Example outputs from our three conditions for a single prompt+topic combination, and the average coherence (coh) and diversity (div) scores for each set of three outputs.

condition	coh	div	One attribute of source code is...
human	4	.38	it is typically written in a human-readable format. editability, so that programmers can easily change it to suit their needs. it is a description a computer program.
custom	4	.37	that it contains code written by humans. its modularity - code modules contain reusable code components. complexity.
baseline	2.6	.08	that it can be used as a source of information. that it can be used as a source of inspiration. its modularity.



(a) Distribution of diversity, split by discipline. Diversity is measured as the average sentence embedding distance per prompt+topic combination.



(b) Mean coherence per prompt+topic combination, split by discipline. Each prompt completion was scored by a domain expert on a scale of 0 to 4.

Figure 5.4: Diversity and coherence measures across three test disciplines for three conditions: a baseline language model, a language model with the custom decoding, and a human-created gold standard. The custom decoding improves upon the baseline and approaches the human gold standard.

ingful cosine-similarities [109], by reporting the average distance between outputs within a given prompt. A higher average distance means that outputs are more dissimilar from each other, and therefore more diverse.

5.4.2 Results

We confirm H1, finding that our system outputs are more coherent and diverse than the baseline, and approach a human-written gold standard. Figure 5.4a and Figure 5.4b show that the custom decoding method outperforms the baseline, but does not reach the performance of the

human-written outputs. We perform two comparisons for each discipline—custom v. baseline and custom v. human—for a total of six null hypotheses per measure (diversity and coherence). For diversity, as we have normally distributed continuous data, we use two-tailed t-tests, and for coherence, as we have ordinal data, we use Mann-Whitney U tests. For each measure, we apply a Bonferroni correction ($m = 6$). We find a significant difference ($p < .001$) for all comparisons. Table 5.2 shows some example outputs from each conditions for a single prompt+topic. These examples demonstrate the quality of the human-written outputs: they are long, detailed, and diverse. Comparatively both language model methods are shorter, less specific, and more repetitive. However, the custom method improves the quality of the outputs.

It is important to acknowledge that the variation in both the diversity and coherence measures are quite large. This means that while on average the custom decoding is an improvement over the baseline, for any given prompt+topic combination the output could be very high quality or of a much lower quality. People using the system will not necessarily see this huge variation; they will only see the 10 or so model outputs that they generate.

We do not confirm H2, that the custom decoding performs consistently across many different topics. Figure 5.5 plots the average coherence for each topic with the black dots, and the coherence for each prompt+topic combination in the colored dots. From this we can see the variation in quality over the topics for the custom decoding method. For instance, the "computer security" outputs score an average of 3.7 in coherence, while "automata theory" outputs score 2.1. When looking at the human-created outputs, the quality is far more consistent, with no topics dropping below an average of 3 in coherence. This demonstrates that our system works well for some topics and less well for others. While we expected that our system would not perform as well as a human would, we did expect that the system would perform more consistently across topics. It is unclear why the language model performs significantly better on some topics, and given the way that these language models are trained it is difficult to inspect or even predict how well the model will perform on a given topic.

However, we do confirm H3, that output quality varies across topic+prompt combina-

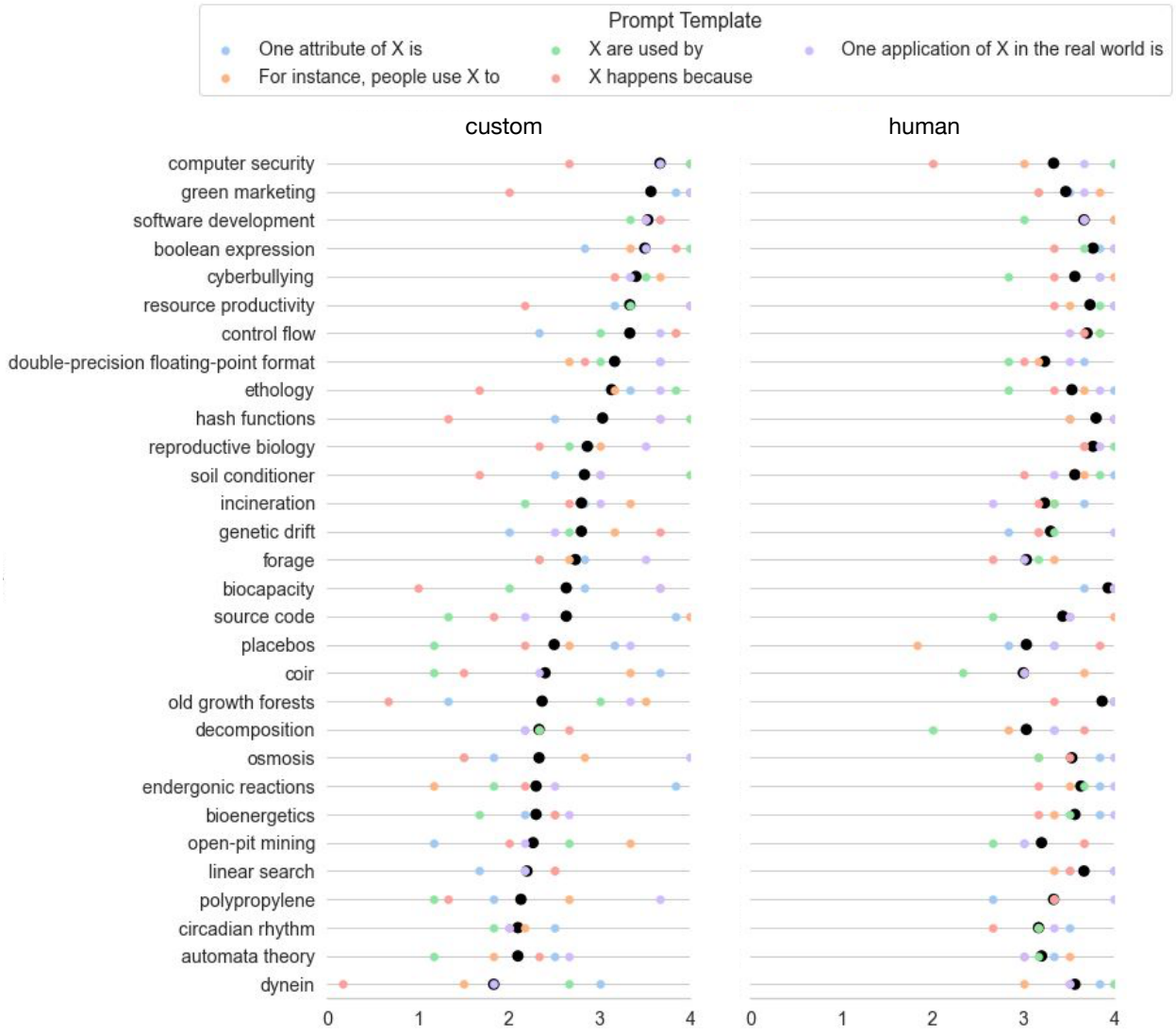


Figure 5.5: This graph shows the coherence per topic for the custom decoding and the human-created gold standard, where 0 is nonsensical or untrue and 4 is generally true. The black dot shows the average coherence of all responses for a given topic, while the colored dots show the average coherence for a given topic per prompt. Topics are ordered by average coherence in the custom decoding. This graph shows that some topics perform much better than others with custom decoding, while the human outputs are generally high quality regardless of topic. It also shows that within a topic there can be a large variation between prompt templates.

tions. Figure 5.5 shows that some prompt templates work better for some topics than others. In the human-written outputs, the variation is smaller, but still we see some range. For instance, let's look at the topic "dynein", the worst performing topic. The prompt "Dynein happens because" scores an almost 0 on the 0 to 4 coherence scale, while the prompt "One attribute of dynein is" scores a 3. Dynein is a family of proteins important in cell behavior. Owing to the nature of what dynein is, it makes sense that the system is more coherent on attributes of dynein, rather than why dynein "happens". However, it's notable that the human outputs scored 3 or above for all prompts for "dynein". Here is a human output about why dynein happens: "Dynein happens because organelles, such as the Golgi complex, need to be positioned in cells." This sentence structure is a little convoluted, but it's clear that the human was able to compensate for the prompt and still write something coherent and meaningful. This highlights the importance of using a prompt that works well for the topic. Since we wanted to test our system with unseen topics, we ensure that participants can add their own prompts in case the template prompts don't work well for their topic.

5.5 Study 2: User Evaluation

The results of Study 1 confirmed that our custom decoding method outperforms a baseline system and approaches a human-written gold standard. In this study we sought to understand how writers make use of sparks when writing, and how spark quality relates to this usage. In particular, we pose the following research questions:

RQ1: In what ways do writers make use of language model outputs?

RQ2: What attributes of language model outputs, if any, correlate with writer usage and satisfaction?

We ran a single condition study intended to stress-test our system with a variety of unseen topics and collect rich data (both quantitative and qualitative) across participant action, perception, and cognition. While we did not have a baseline condition, we asked participants to compare their

experience using sparks to their general writing process. This study was approved by the relevant IRB.

5.5.1 Methodology

Task We evaluated how our system supported graduate students in writing tweetorials. Participants were asked to write approximately the first 100 words (or about five tweets) of their tweetorial.[14]

Participants We use graduate students as they are eager to participate in science writing [110] and many tweetorials are already written by graduate students, demonstrating that this is a writing task our participants may conceivably want to engage in on their own. We recruited 13 STEM graduate students to write a tweetorial on a topic related to their research, while making use of the Sparks system.[15] Information about all participants can be found in Table 6.1.

Procedure The study was run remotely via video chat and screen sharing. Participants were first asked to read an introduction to tweetorials, which explained what tweetorials are and walked through an example tweetorial. They were then introduced to the system and watched a short video that demonstrated the system's features and showed an example use case of the system. Participants could ask clarifying questions to the facilitator.[16] This portion typically took 10 - 15 minutes. At this point the participant was asked to pick a topic to write about, as well as provide a 'context area' that would give context to their topic and aid the system to correctly interpret their topic. Then they were given 20 minutes to interact with the system and complete the writing task.

[14] In pilot studies, participants felt intimidated by having to complete a draft within a specified period of time. By having them write the first 100 words, they were able to fully scope out their tweetorial without feeling pressured to produce a complete draft.

[15] In pilot studies we found that participants did not want to write about a provided topic. Even though topics were selected to be relevant and well-known in their disciplines, participants stated they did not feel comfortable (some said knowledgeable, some said motivated) explaining the provided topic. To encourage a realistic, self-motivated writing scenario, participants in this study were asked to pick their own topic. This had the additional benefit of stress-testing the system on a variety of topics unseen by those involved with the design.

[16] If participants asked to learn more about how the system worked, the facilitator said that it was an algorithm that could generate text in response to a prompt, and that they could discuss the system further after they completed the writing task.

Table 5.3: Participant demographics. Low = once a year or so. Med = Once a month or so. High = once a week or so.

ID	Discipline	Science Writing (general / twitter)	Topic	Context Area
P1	Climate Science	Low / Low	rainfall variability	climate science
P2	Climate Science	Low / Never	predicting climate change	climate science
P3	Climate Science	Never / High	sea level change	geophysics
P4	Climate Science	Low / Low	glacier retreat over the holocene	paleoclimate
P5	Computer Science	Low / Never	computationally hard problems	computer science
P6	Computer Science	Never / Never	pseudorandomness	theoretical computer science
P7	Political Science	Med / Med	document embeddings	natural language processing
P8	Psychology	Never / Low	regulatory fit	psychology
P9	Psychology	Low / Low	motivated impression updating	social psychology
P10	Public Health	Low / Low	measurement of sexism	sociology
P11	Public Health	Never / Never	logistic regression	epidemiology
P12	Public Health	Low / Never	deprivation indices	public health
P13	Public Health	Med / Med	threat multiplier	environmental health

Mouse clicks and key presses while the participant interacted with the system were collected, as well as all sparks generated.

After this, the participant filled out a short survey, which included the Creativity Support Index [111], and partook in a semi-structured interview with the facilitator. During the interviews, participants were asked questions about the usefulness of the system and how their experience differed from their typical writing process. They were encouraged to review what they had written / the sparks they had seen to ground their responses. The survey and interview questions can be found in the Appendix. The entire study took about an hour and participants were compensated $40 USD.

Analysis Participant interviews were transcribed and the authors performed a thematic analysis [112] on the transcripts. The analysis centered on: how sparks were helpful or unhelpful, how writing with the system compared to their normal writing process, and ownership concerns in response to writing with a machine. Relevant quotes were selected from the transcripts and collated in a shared document, where the authors iteratively discussed and collected the quotes into emergent themes. Finally, all sparks seen by participants were collected and annotated for common computer-generated text errors: 'Grammar and Usage', 'Redundant', and 'Incoherent' [113]. These annotations were done by graduate students. The coherence and diversity of sparks seen by each participant was measured as in Study 1.

5.5.2 Results

We structure this results section around our two research questions, and then report on how participants felt sparks compared to existing tools like web searches, and the issues of ownership and agency when writing with a computational aid. Participants came from across five STEM disciplines and selected a wide variety of technical topics to write about (see Table 6.1). We found that participant demographics did not correlate with any of our measures.

RQ1: In what ways do writers make use of language model outputs? Of our 13 participants, nine spoke in great detail about the ways in which sparks helped them. The remaining four reported

Table 5.4: Results of thematic analysis on reasons sparks were helpful. We report the three main use cases. Italics added by researchers to highlight where sparks influenced participant writing.

Use Case	Example Usage and Quote
inspiration	spark: People care about glacier retreat over the holocene because *glaciers affect sea level rise.* what participant wrote: ...Second, *the glaciers in South America have had an outsized impact on sea level rise*. xxx% of the current sea level rise has actually be attributed to the retreat of glaciers in South America! ... quote: "My specialty is very specific and technical. And it's often hard to figure out how to spin things in ways that feel relevant to people who don't study this. Sea level rise is something that people would find relevant."
translation	spark: In sociology, a deprivation index measures *societal conditions affecting individuals' abilities to obtain goods.* what participant wrote: ...relative deprivation experienced by individuals relative to others. It can be defined as *societal conditions affecting individuals' ability to obtain goods*, poverty levels relative to medium household income, among other definitions. ... quote: "Most of the time it [the system] was articulating the ideas that were already in my head in a way that's short and concise."
perspective	spark: One attribute of measurement of sexism is *that measuring sexism involves measuring attitudes towards men versus.* what participant wrote: The researchers in my study wanted to answer the question: "Does the level of sexism somewhere *impact that area's rate of gender-based violence?*" quote: "That was helpful because the research that I do around sexism is not concerned with people's attitudes, and instead concerned about things like incomes or legal rights or education levels. And so I wouldn't have even thought to talk about like sexism as it relates to people's attitudes."

that they did not find the sparks helpful. To answer our first research question we focus on the nine participants who found the system useful. In a later section of the analysis, we will analyze factors that may explain why four participants did not find the sparks helpful. Participants made use of sparks in three distinct ways: for inspiration, translation, and perspective. We talk about each of these in detail. Table 5.4 shows examples of the three main use cases participants reported, which we also discuss in the text below.

First, five of the participants reported on using sparks to provide them with inspiration.

This was our intended use case of sparks, and we call this the 'inspiration' use case. These participants noted that the sparks provided good angles for discussing or introducing their topic. Table 5.4 shows how P4 used a spark about 'sea level rise' to make their topic 'glacier retreat over the holocene' more interesting to the average reader. Similarly, P2 noted that a spark about 'weather prediction models' was a useful entry point to their research on 'predicting climate change'. They said, "that's something within my field that the general public might be more familiar with than what I actually do." P7, writing on 'document embeddings', said, "[the system] definitely generated multiple [ideas] that I could have written different tweetorials about."

Second, six of the participants reported using sparks to help them with translation by providing detailed sentences to start with. We call this the 'translation' use case as participants reported that the sparks helped them 'translate' an amorphous idea in their head into a sentence.[17] Participants discussed the difficulty of writing technical definitions or including technical details, and remarked that although the sparks were often showing them information they already knew well, it was much faster and easier to draw on language from the sparks than to write a sentence from scratch. Table 5.4 shows how P12 used a spark to write a detailed sentence on 'deprivation indices'. They said "that would have probably taken me three sentences to write, then I'd have to spend time editing it down. This is a lot quicker." P7, writing on 'document embeddings', described the utility in this way: "[the sparks] do a really good job of compressing exactly the types of things that I would be going on Wikipedia or Google to get."

Third, three of the participants reported that the sparks showed them external perspectives. We call this the 'perspective' use case, as the sparks showed participants how their reader may be thinking about their topic. Table 5.4 shows how the sparks helped P10, who was writing about measuring sexism. She noted that many of the sparks talked about sexist attitudes and while that certainly is an aspect of measuring sexism, it isn't the aspect that she actually studies and therefore that might be an assumption that she will have to address in her tweetorial. P5, writing about 'computationally hard problems', noted that the sparks contained some technical words like

[17] We borrow the term 'translate' from the cognitive process model of writing [4]

'NP-completeness', which made him reflect on whether or not someone who decided to read his tweetorial may already have some knowledge about his topic. Interestingly, participants discussed sparks that were factually wrong or incorrect in their interpretation of the topic as being useful because the sparks alerted them to misconceptions their readers may hold.

We wanted to investigate how these three use cases correlated with participants' actual interaction with the system. To do this, we labeled each participant with a single use case, where participants who mentioned more than one use case were labeled based on the use case they said was the most prominent or that they discussed the most. This resulted in four participants for 'inspiration', three for 'translation', and two for 'perspective'. (The remaining four participants said sparks were not helpful.) We then looked at writing timelines for each participant, noting when they interacted with sparks. Figure 5.6 shows participant timelines grouped by this categorization. **The 'translation' use case corresponds to much back and forth between writing and interacting with sparks, whereas the 'inspiration' and 'perspective' use cases correspond to longer stretches of independent writing.** We note that participants who said the sparks were not helpful had quite varied interaction patterns, suggesting that interaction pattern alone is not enough to determine utility of a writing support tool.

In order to further examine how interaction patterns differ between use cases, we look at: 1) quantitative measures (the number of sparks generated), 2) temporal measures (the number of times a user swaps between generating sparks and writing), and 3) integrative measures (the average longest common substring between selected spark and what participant wrote). Figure 5.7 shows the results of these analyses. The 'translation' participants requested more sparks and used a higher variety of prompts to do so than others, suggesting that help with translation can occur more frequently throughout this setting of writing, or perhaps that the translation use case requires more sparks as part of its process. Interestingly, the number of starred sparks, as well as the percent of starred sparks (i.e. number of starred sparks divided by total sparks seen) is not noticeably different between the groups, suggesting that different use cases does not mean different levels of usefulness. We also see that 'translation' users moved back and forth between requesting sparks

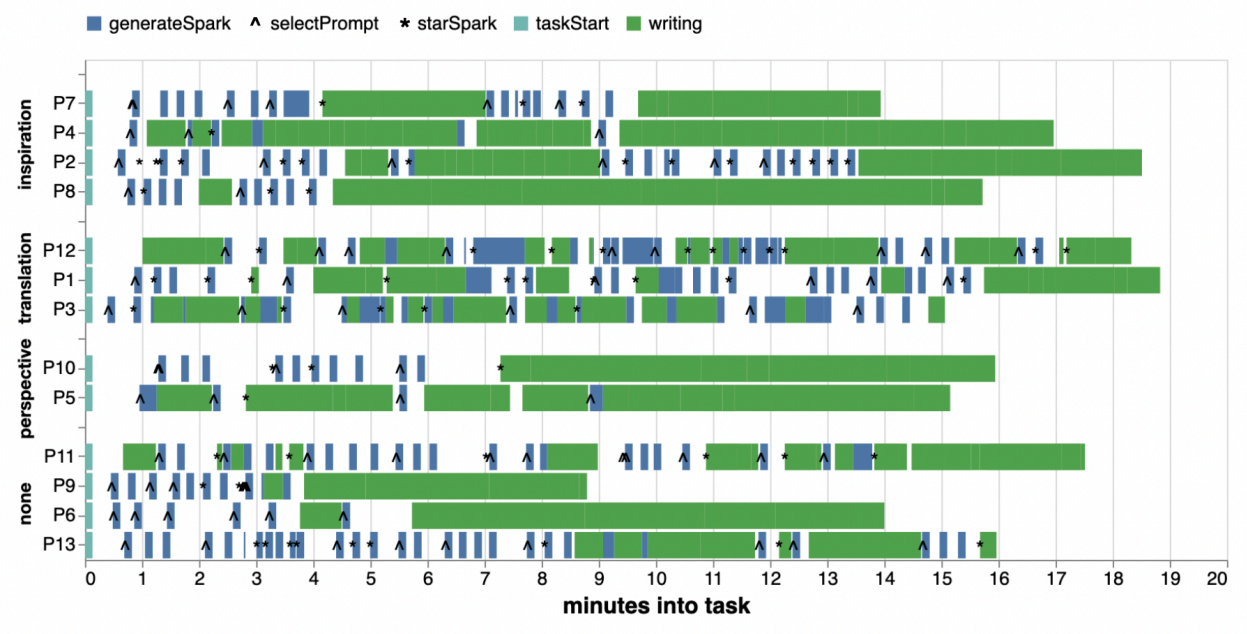


Figure 5.6: Timelines of all participants from the study, with time writing versus time generating sparks marked in different colors. Participants are grouped by their engagement pattern.

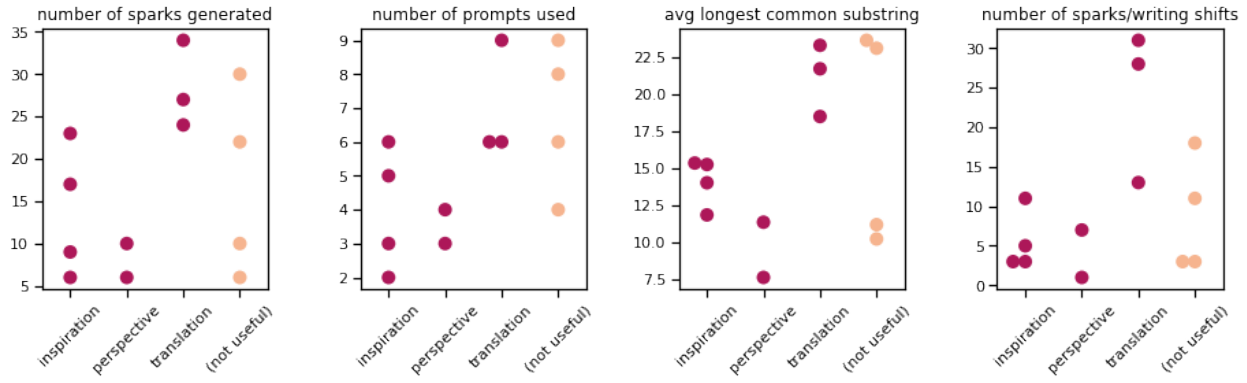


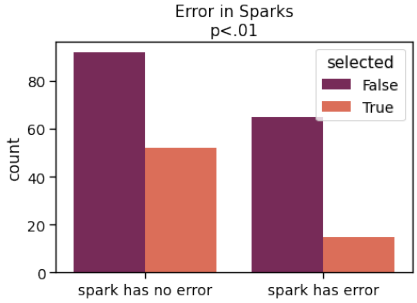
Figure 5.7: Four different measures of interaction, where participants are split by primary use case. Translation users generate more sparks, use more prompts, copy more from sparks, and shift between writing and generating sparks more than other users. There appear to be less differences between inspiration and perspective users.

and writing more often than others; 'inspiration' and 'perspective' users tended to write for longer periods of time uninterrupted. Looking at how sparks were incorporated, 'translation' users tended to copy longer portions of sparks directly into their writing than 'inspiration' users. This analysis shows measurable interaction differences between the different use cases. In the next section, we analyze how the quality of sparks related to interaction patterns as well as participant satisfaction with the system.

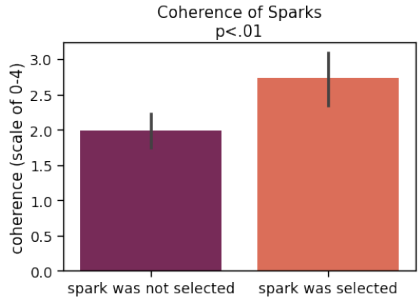
RQ2: What attributes of language model outputs, if any, correlate with writer action and satisfaction? We look at the quality of individual sparks, as well as the aggregate quality seen per participant, and hold the following hypotheses:

- H1: Writers are more likely to star higher quality sparks.
  - H1a: Starred sparks have higher coherence than not-starred sparks.
  - H1b: Starred sparks have less errors than not-starred sparks.
- H2: Writers who see higher quality sparks are more likely to find the system useful.
  - H2a: Higher participant satisfaction is positively correlated with higher average spark coherence.
  - H2b: Higher participant satisfaction is positively correlated with higher spark diversity.
  - H2c: Higher participant satisfaction is negatively correlated with higher average error rate.

Of the 224 sparks seen by participants, 67 were starred, which amounts to 30% of sparks seen. Figure 5.8a looks at the error rate between sparks that were starred and those that were not. Due to sparsity in errors, we collate all the error categories, giving each spark a binary annotation of true or false for whether the spark contains any kind of error or not. Because of uneven sample sizes and the fact that we have a binary measure of error, we use a non-parametric test of proportions, the Fisher exact test, for significance. We find that sparks without errors are significantly more likely to be starred by participants ($p < .01$). Similarly, Figure 5.8b shows the results of the coherence annotation, looking at the coherence of sparks that where starred compared to those that were not.



(a) Sparks without any errors were significantly more likely to be selected ('starred') by participants than sparks with some kind of error.



(b) Sparks that were rated as being more coherent by expert annotators were significantly more likely to be selected ('starred') than sparks that were not selected.

Because of uneven sample sizes, we use the Welch's t-test to test for significance, and we find that starred sparks have significantly higher coherence than those not starred ($p < .01$). **Thus we confirm H1: Writers are more likely to star higher quality sparks.**

To test our second set of hypotheses, that writers who see higher quality sparks are more likely to find the system useful, we look for correlations between our measures of spark quality (coherence, diversity, and error rate) and the results of the Creativity Support Index survey. We look at the individual creativity support measures (expressiveness, immersion, enjoyment, exploration, and results worth effort) as well as the aggregate measure, calculated as recommended by the creators of the index [111]. The aggregate measure nicely matches our interview data, where the four participants who reported that the system was not useful had the four lowest scores. We calculate the Pearson correlation coefficient and p-value to look for a linear relationship between variables and find no significant correlations. **We cannot confirm H2: Writers who see higher quality sparks are more likely to find the system useful.**

The interview data allows us to explore why spark quality may not correlate with usefulness. We can look at how participants who reported different levels of system usefulness responded to the same kind of error in different ways. Let's consider P10, P12, and P13. All are graduate students in the school of public health, and all commented that sometimes the sparks misinterpreted their topic. But P10 and P12 had some of the highest Creativity Support Index scores, and P13 had the lowest. P10 actually saw value in a spark we might consider low quality because it misinterpreted

her topic but gave her additional perspectives she otherwise "would not have even thought to talk about". Describing the utility of sparks P10 said,

> There was a spark about measuring sexism by looking at people's attitudes towards women and men. And so that was helpful because the research that I do around sexism is not concerned with people's attitudes, and instead concerned about things like incomes, or legal rights or, education levels. And so I wouldn't have even thought to talk about like sexism as it relates to people's attitudes.

P12 thought of the spark as human error; they blamed low quality sparks on themselves. They also found the system very useful, and described seeing incorrect topic interpretations quite differently:

> It [the system] kept going to obesity. I think that's because of deprivation... So maybe I put the wrong field [context area]. Like, I could have said sociology instead of public health.

Whereas P13 described the same situation as a system error:

> I think it [the system] saw the word climate change and ... automatically went to the traditional climate change research about, sea level rising and stuff. And that wasn't at all what I was trying to write about.

These participants responded to the same error (misinterpreted topic) in different ways. **This suggests that other confounds, like participant attitudes, make spark quality insufficient as an explanation of perceived usefulness.**

Participants also responded differently to sparks which showed them things they already knew. Some participants found these sparks to be helpful. P1 writing about 'rainfall variability' and who had the highest Creativity Support Index score said, "I was impressed by the accuracy of most of them. [gives example spark] That's awesome. Having that specificity in a concise sense

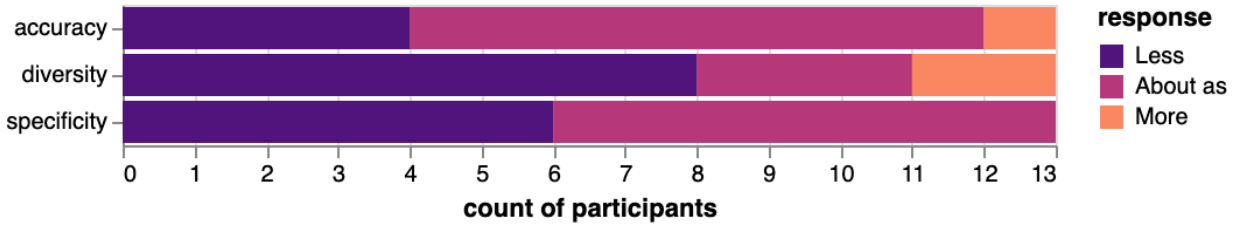


Figure 5.9: Most participants found the sparks about as or more accurate and specific than Google search. This was not true for diversity – most participants found the sparks to be less diverse than a Google search.

was helpful, and more helpful than Wikipedia." On the other side, P9 writing about 'motivated impression updating' and who had the second lowest Creativity Support Index score, found these kinds of sparks to be useless: "I felt like it [the system] would be helpful to someone who doesn't know the topic. Not to someone who knows a lot about the topic."

How did the sparks compare to other resources like web searches? Participants tended to agree that the sparks were about as accurate as Googling, but they varied in whether the system was as useful. Figure 5.9 shows the results of our survey question about how sparks compared to what participants might find via Google. In the interviews participants were able to be more precise about how they perceived the differences. Some said that even though the sparks were not quite as good as Google, being able to stay in the context of writing and not be distracted by the results of a web search was more beneficial to them. Others found the sparks better than Google: P8, writing about 'regulatory fit', said that while Wikipedia is generally a good resource for older psychology concepts, it typically fails for more modern psychology research, whereas the sparks about her topic were correct some of the time.

But several participants mentioned simply feeling more 'in control' when using Google. P2 said, "It's probably just easier to navigate on Google because I'm more familiar with the phrasing and the patterns that will get me the results that I want." P9 said, "I feel like I have full control when I'm googling something over... where my brain wants to go and how I want to think of new ideas." P12 discussed trying different prompts and eventually giving up because they "could not get the prompts to give me that spark [I wanted]". These point to a potential learning curve of

working with the sparks that participants were not able to overcome within their 20 minutes of writing.

Did participants have ownership concerns? Most participants had no ownership concerns about incorporating sparks. Several reported that because the system could never totally surprise them, they didn't feel like it had ownership over anything they wrote. Others said that since they are writing about public knowledge, it was unimportant where their ideas came from. One participant articulated that coming up with ideas is not the hardest part of science writing, but rather putting time and energy into building an audience and writing something engaging, so incorporating sparks would simply be one small part of a much larger endeavor that she took on. One participant compared the sparks to searching on Google (which they did all the time); another compared it to Grammarly (a grammar-checking service). One participant said that the sparks were simply elaborating on his own idea.

However, P9 talked about how he considered outreach and science writing to be part of his job as an academic, and thus any system that automated some aspect of this felt like it was taking over something that he found fundamental to his work. He said, "What this tool is accomplishing is an end in and of itself, right? Getting the opportunity to practice these things [idea generation for science writing] and organically generate them for myself is part of what it means to be an academic for me." P9 was also one of the participants who did not find the system useful.

While most participants had no ownership concerns, all participants expressed concerns about plagiarism. Several participants brought up that they were unsure exactly where the sparks were coming from, and they wanted to make sure that anything they took from the sparks was adequately changed, to alleviate any concerns about plagiarism. P2 described this as, "I think if I was using something like this, I would probably never use an entire sentence verbatim. Just because, if you don't know where it's pulling it from... I wouldn't want to run the risk of plagiarizing something accidentally even."

5.6 Discussion

5.6.1 Why do some people find AI assistance more useful than others?

We found that there was no correlation between the average quality of sparks seen by a participant and how useful that participant found the system.[18] Several other studies of computational aids in a variety of domains have also found a high variation across participants in how useful a system is [116, 3, 97, 117]. In this section, we consider what else might be impacting perceived usefulness in human-AI collaboration.

The idea of an objective 'quality' of a system may be misleading. For example, presenting random words may seem like a reasonable baseline that more sophisticated systems can improve on. But randomness can be quite a strong baseline when it comes to creativity support. For instance, singer-songwriter David Bowie famously used random text generators when writing song lyrics; he used a tool called 'The Verbasizer', a computerized version of the cut-up technique which dates its history back to at least the Dadaists in the 1920s.[19] There exist today thriving communities—e.g. experimental writing, electronic literature—that draw upon surrealism and computation, and regularly makes use of randomness as a form of writing and/or writing support. For instance we can consider the contemporary work of John Cayley, Lillian-Yvonne Bertram, and Alison Parrish as grappling with the role of randomness in writing.

But what drives some writers to partake in this exploration? Probably since the beginning of time some creators have been seeking out inspiration in whatever form was available to them, and others have not. It might be that people's attitudes towards influence and inspiration has a large impact on their attitude toward a computational system, perhaps moreso than the quality of the system itself. This would explain why random suggestions can be seen as very useful by some,

[18] This isn't to dismiss the impact of quality: within a participant, they preferred high quality sparks. And the proliferation of writing tools that make use of generated text [23, 114, 115] is likely due to the increased quality of generated text, and its correspondingly increased usefulness. But it seems like there are other confounding factors that complicate the relationship between system quality and perceived usefulness.

[19] Vice wrote an article about The Verbasizer in 2016: `https://www.vice.com/en/article/xygxpn/the-verbasizer-was-david-bowies-1995-lyric-writing-mac-app` and a modern version of the tool is available: `https://verbasizer.com/`

and factually correct generated sentences about a technical topic can be see as useless by others. Currently, it's unclear how people's openness to other kinds of influence, like random inspiration or ideas from a mentor or peer, relate to their openness to machine influence.

Expectations may also play a large role. One strength of large language models is writing sentences that we already kind of know—this is seen through the success of Gmail's Smart Compose, which seeks to only suggest extremely likely sentence completions [32]. But even that was divisive in our study, in which some participants appreciated sparks that detailed what they already knew (it helped them write concise, technical sentences more quickly and allowed them to stay in the context of writing) while others reported those kinds of sparks as useless. People may bring in expectations of the *kind* of help they are looking for, and dismiss anything that doesn't fit their model. Others may be more open, even looking for ways to find the system useful in the face of unexpected outputs.

Overall, we believe that participant attitudes are a major unknown factor when studying human-AI collaboration. Future work should investigate, or at least acknowledge, this confounding factor, as it complicates the seemingly simple question of 'how useful did you find the system'.

5.6.2 Providence and plagiarism as major writing concerns.

Most participants were worried about providence and plagiarism, bringing up these issues independent of any prompting from the interviewer. They didn't fully understand "where the sparks came from" and were worried that copying too much would be considered plagiarizing. This is not a concern we've seen reported in work on human-AI writing collaboration previously. Research on language models is attuned to if the model is copying from the data it is trained on [20], because that is viewed to be a sign of a low quality model and can result in data leakage from the training corpus. But even if we assume that the model is not copying from the source data, we may still need to ask the question of if it is okay—or even possible—to plagiarize from a language model.

Dehouche raises the ethical issue of plagiarizing from GPT3, stating that while language models have long been used for plagiarism *detection*, there needs to much more inquiry into plagia-

rizing *from* the model now that they can generate much more coherent, long-form text [118]. Dehouche argues that GPT3-generated text raises basic questions about authorship, because the author could be conceivably be the person who prompted and supervised text generation from the model, the computer scientists who developed and trained the model, the company offering access to the model, or the various anonymous authors whose text makes up the training data of the model.

In our setup participants would have struggled to plagiarize a whole tweetorial[20] yet they still raised these concerns. Historically plagiarism has assumed there is another scholar from which to steal words or ideas [119], but since authorship of text generated from language models is unclear, the issues of plagiarizing from them are unclear as well. Presumably the assumption of commercial writing support systems is that the writer is also the author of the generated text, thus removing any concern of plagiarism, but we saw that was not the assumption in our study. More work is needed to investigate this important question.

5.6.3 Bias in language models and the value of a biased perspective.

The bias of large language models is well-documented and a serious concern for anyone making use of this technology [19]. We selected the task of science writing as one where we expected there to be minimal issues of racism, sexism, and other kinds of prejudices brought up during the task. However, we still saw that the model was biased towards more prevalent topic associations. We saw this particularly in the case of sparks that misinterpreted a topic: these sparks were not wrong per se, but responded to the prompts with a viewpoint which sometimes differed from the participants' particular line of inquiry. Smaller language models trained on a more carefully curated dataset seem like a good solution to this problem, though it negates the utility of multi-purpose models.

Participants who reported these incorrect interpretations as useful introduce a novel use case of bias in language models more generally. If we acknowledge that language models are inherently biased based on their training data, we can start to envision how we might make use of that knowledge. For example, Schmitt and Buschek use chatbots as a way for story writers to develop

[20]though commercial systems like `https://rytr.me/`, `https://researchai.co/` and `https://www.sudowrite.com/` will happily spit out whole essays or stories

characters, where writers progressively turn a bot into a specific character [120]. A biased language model is providing a specific perspective, and writers could make use of that perspective as a way to imagine their reader. Imagining your reader is an important and difficult part of writing [4]. What knowledge does your reader already have? Where will your reader get confused? When does your reader get bored? Great authors constantly consider these questions and adjust their writing accordingly.[21] Biased language models may be able to help writers model their reader, and help keep writers aware that any language model contains some kind of bias.

5.6.4 Limitations

Our system used a specific language model with a specific prompting method. Available language models are changing rapidly, as is the research on how to best make use of them. And while we picked our task to be representative of constrained and creative writing tasks, it differs greatly from other writing tasks people might be interested in like writing stories, academic papers, newspaper articles, or marketing copy.

Because we wanted our user study to closely mimic a realistic writing scenario, we had participants select their own topics. However, this introduced a large confounding factor, as different topics are more or less difficult to explain and make interesting, and different topics may elicit different levels of spark quality from the system, as seen in Study 1. One way we dealt with this confounding factor was by performing a single condition user study, as it didn't require us to control topics across conditions (and therefore across participants). This also allowed us to stress-test the system across many different, unseen topics. However, future work could benefit from comparative studies, either large-scale ones where participants can still pick their own topic but the size of the study minimizes topic as a factor, or smaller-scale ones where participants are assigned topics.

The small sample size of our study may have limited our ability to find significant correlations. Perhaps in larger studies we would find that the quality of system outputs *does* correlate with perceived usefulness. A hypothesis we hold, which would need to be tested, is that quality impacts

[21] Novelist George Saunders discusses this in an article for The Guardian: `https://www.theguardian.com/books/2017/mar/04/what-writers-really-do-when-they-write`

perceived usefulness up to a point, after which increased quality has less impact than participant attitudes. We hope the results of our study inspire future work that can continue to explore how writers interact with language model outputs.

Chapter 6: Social Dynamics of AI Support in Creative Writing

In this chapter I report on qualitative research investigating when and why a writer might turn to computational support. This work is in direct response to my prior work with Metaphoria and Sparks, in which I built and evaluated the use of systems to support constrained, creative writing. While in all my studies a majority of writers found the generated text coherent and useful, there was still a distribution of responses to the systems.

In Metaphoria, some writers felt threatened by a system that could produce "an amazing line of poetry", while others found that high quality outputs allowed them to focus on other goals, considering Metaphoria more like a "a calculator for words". In Sparks, I similarly found that the correctness of the outputs did not correlate with writers wanting to incorporate them. While some writers thought any kind of support was useful, thinking of the system as a more useful version of Wikipedia or Google even when it made mistakes or didn't match their expectation, others felt the system took away an important part of the writing process. In the study of Sparks, some writers even found use in incorrect sparks, creatively interpreting them as representing a reader with misconceptions, while others saw incorrect sparks and lost trust in the system.

What explains these differences in responses to the same system? The difference did not seem to lie in the what the system generated for the participants, but rather something about the participants' attitudes towards computational help. This chapter reports on interviews with creative writers about support, in particular how they feel about different kinds of influences—from peers and mentors, but also computers. It addresses this question of why writers respond so differently to the same system, and outlines important factors that modulate writers' experiences with such systems.

This work studies the social dynamics of when and why creative writers request support, whether that support comes from a peer, mentor, or computer program. Though we study a specific

endeavor—creative writing—our work has implications for all kinds of complicated tasks and the role of technology in pursuing them.

I consider how large language models might fare as writing support tools given the social dynamics of requesting help with a creative writing project. I ask the following research question:

> *RQ: When and why might a creative writer turn to a computer versus a peer or mentor to provide support?*

I interview 20 creative writers from a variety of writing genres. The interviewees include 6 creative writers currently using a generative AI creative writing support tool. The interviews focus on what influences their writing practice, first asking about existing kinds of influence such as suggestions from peers, then asking about hypothetical computer programs that could provide human-like support. This work builds on existing frameworks of writing cognition and creativity support, uncovering new dynamics which modulate user responses to technology. Through a qualitative analysis we discover three elements that govern a writer's interaction with potential support actors:

- what writers *desire* help with,
- how writers *perceive* potential support actors, and
- the *values* writers hold about the writing process.

The results align with two existing models. First, the types of support desired can be aligned with the updated cognitive process model of writing [5], which includes motivation and goals. Second, we build upon the support relationship types proposed in Chung et al. [121], contributing the organizing principles of a) how an artist perceives a support actor, and b) how an artist's values impacts when and where they turn for support.

Finally, this chapter discusses how these findings reveal when and why writers might turn to computers for support, and outline future work for building rich interactive writing support tools.

6.1 Methodology

6.1.1 Study Procedure

Between March and August 2022 we interviewed 20 creative writers about their writing practice and their attitudes towards hypothetical computational language technologies. The interviews were focused on:

- the interviewee's existing writing practice (e.g. "What is a piece of writing you are very proud of? Why?"),
- their existing modes of influence (e.g. "Are there people who currently influence or in the past have influenced your writing? Who? In what ways?"), and
- their response to hypothetical computational tools that could act 'at the level of a peer' (e.g. "If a computer program could suggest places to revise like a teacher or peer could, would you use it? Why or why not?").

Some interviewees had experience with an existing creative writing tool called SudoWrite[1]. SudoWrite is a piece of commercial software that requires a monthly subscription and is marketed primarily as a story writing tool. It uses a large language model as its underlying technology. SudoWrite provides capabilities such as continuing a story where you left off, describing a scene, rewriting according to some guidelines, brainstorming plot points, and feedback. We provide examples of SudoWrite's capabilities in Appendix C.1.1. Writers with experience with SudoWrite had a modified version of part 3 for the interview that focused on their use of and response to SudoWrite, instead of hypothetical tools.

The interviews were semi-structured; the interviewer asked follow-up questions or skipped some questions in the guideline when appropriate given the content and context of the interview. Questions were also altered to best probe the writer's genre. The guideline used by the interviewer can be found in Appendix C.1.2.

Interviews were conducted via video chat, were conducted in English, and lasted about an

[1] `https://www.sudowrite.com`

hour. Participants were compensated $50USD for their time. The study was deemed exempt by the relevant ethical review board.

6.1.2 Participant Recruitment

Definition of Creative Writer We recruit anyone who identifies themself as a creative writer. We believe the definition of an 'expert' or 'amateur' creative writer is difficult in a field that has unclear professional delineations. Many successful writers retain full-time jobs as teachers, editors, or in unrelated professions, as few are able to make a living from their writing alone. One potential way to screen participants is to select only participants published in certain venues (this approach is used in [122]). However, using publication by a major publishing house as a metric for expertise will continue to enforce the marginalization of many writers, as major publishing houses repeatedly fail to diversify their writers, editors, and leaders [123]. Another would be to recruit only those with a formal creative writing education (e.g. a Masters of Fine Arts in Creative Writing). However, the cost of these programs can be preventative for many people, and analysis has found that novels written by people with an MFA are not detectably different from those written by people without one [124]. Given these concerns, we recruit widely and allow participants to identify themselves as creative writers. This resulted in a range of participants, from tenured professors of poetry with three critically acclaimed books to writers with only informal education currently submitting their first novel for publication.

Sampling Method We used purposeful sampling for maximum variation because we wanted to identify information-rich cases with the aim of "capturing and describing central themes that cut across a great deal of variation" [125, Chapter 5]. Given the context of creative writing, we wanted to capture insight from a variety of writing genres, as we expected that different genres may have different aesthetic and personal concerns. For example, a fantasy writer may have different concerns about ownership and voice than a memoirist or poet. In addition to this, we recruited participants who use an existing writing tool, SudoWrite, that provides human-like responses to writing.

This allowed us to gain insight into the dynamics of those currently using a large language model-based support tool. We continued recruiting until we had adequate variation in participants, and saturation of themes.

Recruiting Procedure Participants were recruited through MFA graduate school distribution lists, professional networks, and personal contacts and writing communities. For recruiting SudoWrite users, we reached out to writers who had written about their experience with SudoWrite online, either via personal blog posts, published interviews, or social media. We then used snowball sampling, asking those we found to introduce us to other SudoWrite users.

Participant Population Table 6.1 reports the genre and writing education of all participants. The demographics of the participants were: 8 women, 7 men, 1 non-binary, 4 undisclosed; 7 aged 18-25, 5 aged 26-35, 2 aged 36-45, 2 aged 46-55, and 4 undisclosed. Most SudoWrite users that were recruited were novelists, in line with who the tool is marketed toward.

6.1.3 Analysis and Coding

All interviews were transcribed using an automatic transcription software that kept the original audio aligned with the transcription. We used a general inductive approach for analyzing qualitative data [126] because we wanted to develop a framework of the underlying structure of writer's support experiences. Following this method, two researchers independently read all of the transcripts and identified relevant segments of text. Each researcher assigned each segment of text an initial potential low-level category. Then, through repeated discussion, the researchers reduced category overlap and created shared low-level categories. Finally, these low level categories were collected into high level categories. The researchers repeatedly met and iterated to construct these categories; during the later meetings a third researcher was present to give further insight into the data. This analysis was started concurrently with the interviews being conducted, such that participant recruitment continued until a saturation of themes was found.

By the end of the analysis process, all relevant text segments were collected and annotated

Table 6.1: Background of Participants. W prefix in ID stands for 'writer'; S prefix stands for 'SudoWrite user'.

ID	Genre	Education
W1	Non-fiction	Some classes
W2	Science journalism	Science writing workshops
W3	Poetry	MFA in poetry
W4	TV comedy scripts	Some classes
W5	Fiction, novels	MFA in fiction
W6	Poetry	MFA in poetry
W7	Poetry	MFA in poetry
W8	Fiction, poetry	MFA in fiction
W9	Analysis essays	Informal
W10	Poetry	MFA in cw
W11	Historical fiction, science fiction	Some classes
W12	Poetry, essays, short stories	Some classes
S13	Paranormal cozy mysteries (novels)	Part of ug in English
W14	Personal essays	Student news rooms
W15	Creative non-fiction, flash fiction, poetry	Mostly informal
S16	Novels (fantasy, urban realism, magical realism)	Some classes
S17	Historical fiction	Informal
S18	Fantasy stories	Some classes
S19	Fantasy and science fiction novels	Informal; online communities
S20	Young adult science fiction novels	Some classes

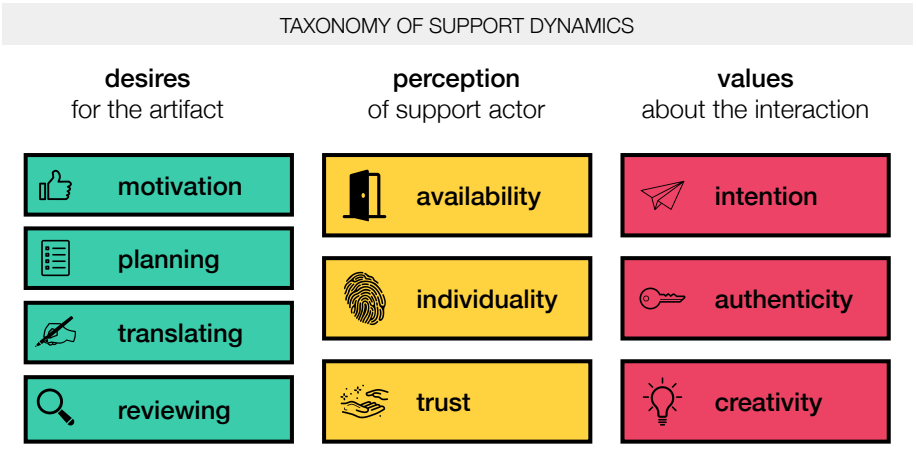


Figure 6.1: Results of qualitative analysis.

with one or more low-level categories. Although our goal is not to make claims about the relative importance of different categories in creative writers as a whole, we do report the prevalence of different categories in the interviews to provide insight into their occurrence in our interviews.

The construction of categories from the data was driven by the research objective to understand what impacts creative writers' desire to interact with and be influenced by computational writing tools. After the categories were consolidated, the researchers considered how the categories relate to models of writing [4, 5], and theoretical work on creativity support [121]. This consideration did not influence the creation of the categories, but rather constituted further analysis into the relation between categories, and the meaning and implication of the results.

6.2 Results

Figure 6.1 shows our three high level categories: writer desires for support, writer perception of a support actor, and writer values about the writing process. Each category contains 3 or 4 sub-categories that delineate what impacts the high-level categories. Our results are not intended to define creative writers' aggregate attitudes toward help seeking or computational tools, but rather account for what impacts their attitudes. Table 6.2 gives a definition for each category, and exemplary quotes.

Table 6.2: Code Description and Example Quotes.

Code	Description	Quotes
Writer Desires for Support		
Planning	Coming up with ideas, plotting, deciding on what to work on next.	"I always struggle with plot and endings. I have even tried to use a plot generator. They've been actually not that great."
Translation	Figuring out how to 'translate' ideas and thoughts to words on the page.	"My career is based on speed. The faster I write, the faster I can get it out, the more money I'm going to make."
Reviewing	Getting feedback, making edits or identifying parts that need work.	"I want other people's perspectives because mine is limited, and people can see things in my poems that I don't necessarily see."
Motivation	Getting affirmation, keeping up motivation (on a specific project or in general).	"Often it's at the end of a writing day and I would like to acknowledge that I was productive and I'm excited about something I'd like to share."
Writer Perception of Support Actor		
Availability	The availability of an actor, for instance an actor must be both physically available (e.g. not asleep), and socially available (e.g writer may feel they've already asked for help too many times).	"Giving feedback takes time and work and I don't want to be asking people to do that work for free all the time." "The advantage that SudoWrite has is availability, because you can't just walk up to a person at any time of the day or middle of the night and go, Hey, I have this idea. How about this?"
Individuality	The actor has individual characteristics, such as aesthetic preferences or lived experiences, that modulates the kind of support they provide, and how the writer views any suggestions.	"Every commenter has a perspective, and you understand what they bring to the table. You'd have to develop that about the machine."
Trust	The actor must be trusted, for instance to have relevant expertise or to deal with sensitive or personal topics.	"I respect her skills as a writer, and I trust that she knows me well enough to know what's trying to happen; ... she's a person who I've given full trust to in many ways and I'm willing to give that trust here."
Writer Values		
Intention	Writers have intentions or goals that an actor may or may not respect or even understand.	"It's not like the computer can understand what you want to say, they can only see what you have written." "But obviously, you have a reason why to write a story. And I think that is something SudoWrite can't reproduce."
Authenticity	Writers have different ideas about what is required to maintain authenticity, and this modulates when and how they want an actor to influence their writing.	"I would be fearful that I wouldn't sound authentic. It's the same reason that I don't really believe in ghostwriting." "I feel like it would feel maybe a little bit creatively dishonest if a computer wrote the ending to my poem for me."
Creativity	Writers have different ideas about where creativity lie, and this influences when and how they want an actor to provide support.	"Computers can [only] help us understand or generalize what is the well-trodden path." "[SudoWrite] had introduced a completely new character, and gave him two stanzas of a song that he was singing, it completely wrote the song. And I was floored."

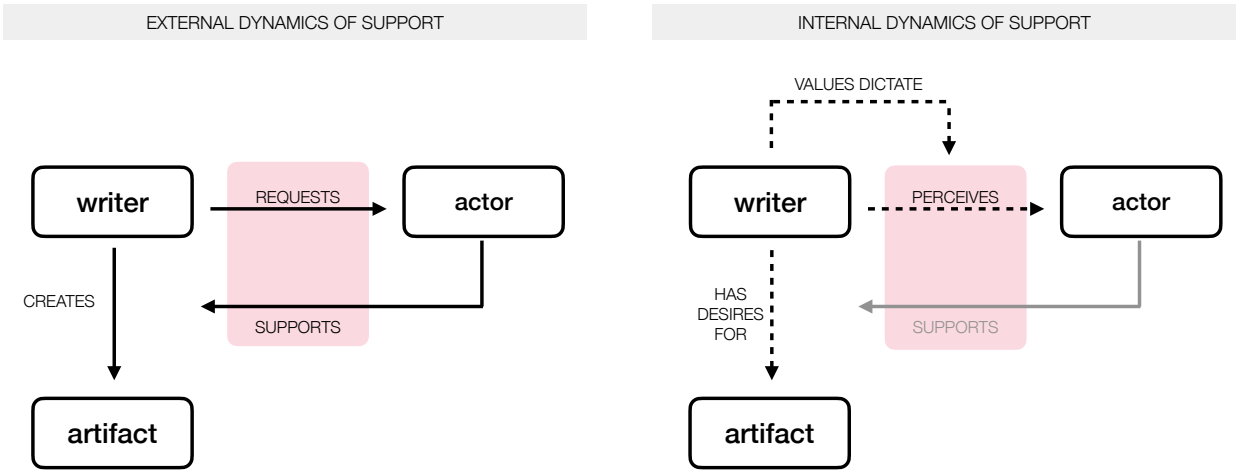


Figure 6.2: External and internal dynamics of support.

6.2.1 Model of Social Dynamics

We cast our results in the similar terms as Chung et al. [121], which proposed a model of relationship types in artist support networks. In their model, they consider three entities: the **artist**, the **actor** (who provides support), and the **artifact** (whatever the artist is trying to create, in our case a piece of writing). We replace artist with **writer**, to make clear our participant population. In Chung et al. [121], the entities are related by how they exchange ideas and contribute to the implementation of the artifact. Additionally, they consider only human actors. In contrast, our work considers the dynamics of why and when the artist turns to an actor for support, and considers a computer program, as well as a person, as a potential actor. However, at heart we also seek to model the relationships that occur when artists seek support, and see our results as an extension of their model.

In Figure 6.2 we show a diagram of how the three entities relate to each other given our results. On the left, we see the external dynamics of support. A writer creates an artifact; a writer requests help from an actor; the actor provides support for the creation process. (These actions are not necessarily linear.) Highlighted in red is the request/support dynamic, which is what we are investigating in this study.

On the right of Figure 6.2, we see the internal dynamics of support, which we report on in

this study. The writer doesn't just create the artifact; they have desires for the artifact, and the actor supports the writer in achieving those desires. The writer also perceives the support actor, and this perception modulates how they choose between different kinds of support. The writer has values about writing more generally, and these values dictate what they want out of the support relationship. In this way, the writer's values impact the kind of support they seek out.

This model is intended to make concrete what goes on 'under the hood' when a writer is looking for help. We call these the social dynamics of support because they model the interactions between individuals—namely, the writer and support actor—and these interactions are the source of aggregate-level behavior, such as a support tool being adopted by many writers.

In the following sections, we report in depth on the themes that emerged from our interviews, and how they together lead to a better understanding of when and why writers look for support.

6.2.2 Writer Desires for Support

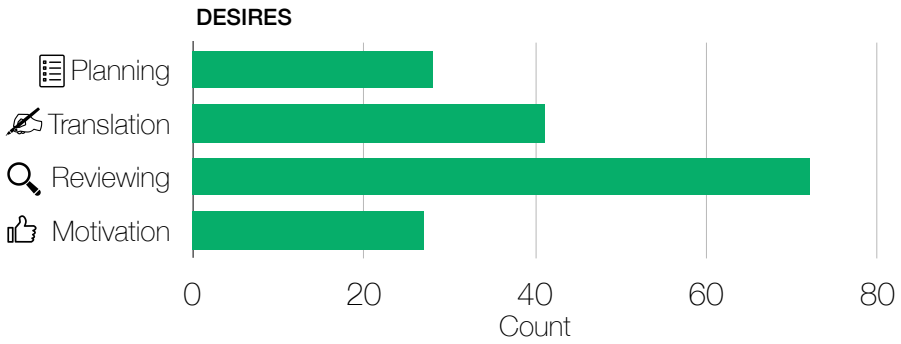


Figure 6.3: Prevalence of codes in data.

Planning Writers often sought support for planning. The kinds of planning spanned from early-stage brainstorming before anything had been written, to how to tie up a poem or some plot points. Some talked about wanting open-ended inspiration, while others wanted help with research or coming up with details. S20, working on a science fiction novel, described having a great brainstorming relationship with her teenage son, who also loves science fiction and enjoyed swapping plot ideas.

Many writers talked about doing research as part of their writing. W10 described researching starfish anatomy; W11 talked about researching social mores in the medieval period; W5 talked about referencing a list of slang from a certain historical period. While some writers were hesitant to use a computer for fact-checking, others were open to a computer program that was "kind of like having a research assistant".

In the interviews we explicitly asked about a hypothetical program that could help with endings. We wanted to understand how writers felt about computational support not just at an early stage, which much research has focused on, but later on in the process. Writers were mixed. W11 said they weren't against such support, but that it would take "the pleasure out of writing. Pleasure is finding the solution to the problem." In contrast, W8 said suggested endings would be really useful because either she would use a suggested ending, or realize that those endings wouldn't work.

Translation The process of getting words onto the page can be a difficult part of the writing process. Whether it's to start an essay, continue a scene, or just select the correct word for what they wanted to express, all writers discussed the difficulty of translation. However, no writers talked about turning to other people for support in this part of the process. Instead, writers curated their own techniques to get them to write, whether it was to literally "draft standing up" (W2 drafted at a standing desk), turn to thesauruses (W8 said "sometimes I'm really frustrated because I feel like the words I'm using don't have the right texture or sound"), or turn to a computer program.

Several writers, both those who used SudoWrite and those who didn't, discussed using a computer program to help them with writer's block. W14 described a hypothetical situation, "where I'm not sure how to proceed, but [could use the computer] to see some possibilities laid out." S16 talked about how "it's a lot easier to react to something and make modifications than to come up with something from nothing." S20 talked about how SudoWrite brought ease to her writing, saying she could "pound her head against the wall" and "open up other books and flip through for inspiration", but SudoWrite eliminates this struggle for her when she feels stuck.

Reviewing When talking about how they typically sought out support, writers mostly talked about getting feedback on their writing.[2] Whether they got feedback from a best friend or a professional editor (or, for the lucky few, someone who was both), the ways they talked about feedback were extremely similar, despite our participants coming from a range of genres and writing education. Almost all discussed the importance of needing other people's perspectives, whether on a poem (W10 said "people can see things in my poems that I don't necessarily see") or a newspaper article (W2 said "I was mostly interested in how it was working structurally, and the clarity of the analysis").

Writers expressed the importance of specificity in the feedback. W15 describes how if a reader says, "this might be a bit boring," he may agree but not have enough information to know how to not making it boring, whereas if they said, "this [particular sentence] is where I lost the thread", that would help more with the editing process. W5 described this occurring when his friend gave him some feedback on part of his novel. He describes the situation: "She let me know that it felt contrived to her. And we were able to even pinpoint what language felt contrived to her so that I could then rework it and smooth it into a way that felt more organic." In this situation, W5 had an extensive back and forth with his friend, which resulted in much more helpful feedback.

In this way, writers expected an explanation for feedback, especially if it came from a computer. W5 said, "I would want to be able to interface with the program to understand why it thinks it needs revision." He gave this example of the level of specificity he imagined being incredibly useful: "If it could say, in the whole body of your text, you've only used natural images in your similes. This is an unnatural image that you're drawing from—do you want to have one that breaks that pattern?" W7 similarly wanted to know what the computer was reading for, saying "Maybe if the computer was delineated as reading for technical or imagistic… if I knew what lens it was reading in. Otherwise I'd be like, 'based off of what?'"

Feedback is often laden with implicit value judgments. W1 discussed the situation of a com-

[2]Two writers, W3 and W6, rarely sought feedback from peers. Instead, W3 saw successful publication as a kind of useful feedback. W6 talked about how getting lots of feedback was one way to write, but it wasn't the way he was writing.

puter giving her feedback that a certain passage is boring, saying "I would be so angry!" because, for example, sometimes the point of a book may be to explore plotlessness. Specificity or being able to question feedback is a way to move past value judgements into more concrete territory. We come back to this idea of value judgements when discussing writer's intentions.

Motivation When writers talked about how they were influenced by other people, or the kind of support they tended to seek out, they often talked about motivation or affirmation as an important part of the writing process. The writers talked about how writing, especially writing a large project like a novel, was a vulnerable, ambitious, and often lonely activity. W4, who was working on the script for a TV show pilot, said, "If I don't periodically get validation? The operation is a bust."

The importance of affirmation ran across most of the writers we interviewed. Writers talked about having no idea how their writing would come across, or wanting something "to confirm it's not trash". S20 described the first time someone else talked about the characters in her novel as if they were real people, and the importance of this indication that what she had created (i.e. the fictional world) was legible to others.

Writers described that when they're looking for support, it can be useful to know or be explicit about if they are looking for critique or validation. As W7 explained, "A lot of times you want validation, but you asked for feedback, or you want feedback, but you're asking for validation."

6.2.3 Writer Perception of Support Actor

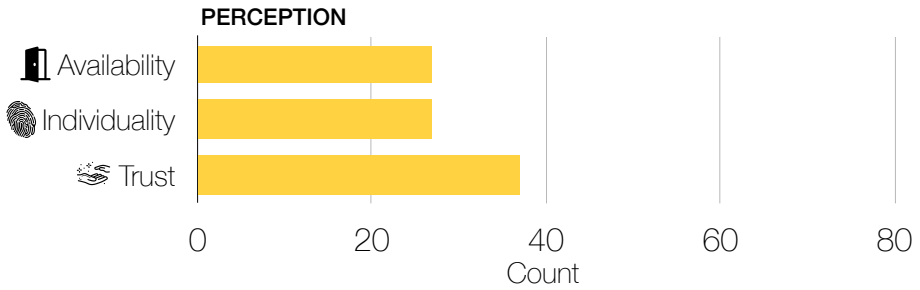


Figure 6.4: Prevalence of codes in data.

When writers talked about people who influence their writing, they talked about the particulars

of these people. Some talked about their spouses; others talked about family members (parents, siblings, children). There were old high school friends, writing workshop peers, and online writing communities that shared niche interests. When discussing these people, they discussed not just their social relationship, but their perception of how a person might provide support. This perception was key for understanding when and why they might ask a person or a computer for support.

Availability The availability of a support actor was always on writers' minds. Many writers noted that requesting help with their writing was an imposition. By asking for support, they are asking a friend or acquaintance for time, and were always mindful of such a request. W12 preferred reciprocal relationships, where she would also be helping the other person, saying, "If it's not this back and forth, like we're both interested in writing and both giving feedback, then I don't really feel comfortable asking for that favor." Two writers had partners who were heavily involved in their writing. W2 noted, "I'm fortunate to have a partner who's available to offer feedback in a reasonable amount when I'd like it. So I have an in-house editor... Without that, I'd feel much more adrift." But even W5, who talked about his fiance as an important figure in this writing process, noted that her time was limited: "If she had the time, I would ask her to edit the whole book, but she has a full time job."

Several writers noted that a computer program wouldn't have these issues. S19 said, "The advantage that SudoWrite has is availability, because you can't just walk up to a person at any time of the day or middle of the night and go, Hey, I have this idea. How about this?" Even non-SudoWrite users noted that making use of a computer program wouldn't be an imposition on the computer. W14 said, "I'm imagining things [to request from a computer] that feel too mundane somehow to ask someone for their time".

While computers were understood as being always available, writers didn't necessarily want replicas of their not-always-available peers. W2 said that because he does have access to peers, a program that replicated his peers' response would thus be uninteresting. Instead, he would want

a computer program to give him something different, something that wasn't currently available to him.

Individuality When writers discussed a support actor, the details of the actor were important. Not all people were the same, and the individual characteristics of a person (or computer program) impacted not only who they turned to for support, but what they did with the support provided.

One important feature they considered was expertise. W2 talked about how a scientist may be relied upon to point out a factual error, but may not be trusted to critique the quality of an opening paragraph. W9, on imagining a computer reading her work, said, "I would also need to know the reading background. Is this a high school, college, PhD student? What is their level of experience of the topic at hand, so on and so forth? Are they a skeptic or optimist? There's a lot of things to consider."

Writers would come to learn specific characteristics of people that would modulate when they'd turn to these people for support. For instance, W8 discussed how the life experience or writing interests of a peer would dictate who she would send it to, saying "Since my brother is also Indian, if I want to know how something reads to another Indian person, I will show him. But then if I'm writing a story about girlhood, I'll send it to my friend Jen, who also writes about girlhood." S19 even described situations where negative feedback may indicate he's on the right track, saying "I can kind of place their feedback into preference categories... there's certain aspects where I know if a certain person doesn't like that, then it's exactly what I want to achieve in that part of the story."

When discussing a computer support actor, several writers talked about the impossibility of a "universal" reader. W10 worried that computers would represent only a dominant perspective, saying, "The 'universal' perspective has been the perspective of cis straight white men and any other perspective is just not considered universal." Others noted that, based on their understanding of how such a computer program might work, it would reflect generalizations of its training data, and lose the individuality that people provide. W6 described the uniqueness of humans in this way: "Let's just say there's a 1%, that is unpredictable, a response they'll have that does not fit the

pattern . . . I'm interested in that 1%, too. I like the inherent unpredictability of a person."

SudoWrite users were able to articulate the unique characteristics of SudoWrite. S20 describes their sense of SudoWrite:

> A peer is someone who is grounded in a very specific point of view, and culture and identity and preference, you know, their own reading habits and a peer can be a very valuable partner . . . when I turn to SudoWrite, I know that I'm getting feedback and interactions with my work that is not personal at all . . . The amount of information on hand that SudoWrite is pulling from is this vast trove. And that's something that a human could never, even if they're well read, could really never achieve.

In this way SudoWrite presented a fundamentally different kind of individuality than a person. SudoWrite is more general, but also more well-read and capable. Still, SudoWrite users discussed the strengths and weaknesses of SudoWrite as they might for a person. For instance, S13 talked about SudoWrite's excellent ability to write descriptors into a scene, but noted that it's terrible at generating humor.

Trust When discussing when and why writers wanted support, trust repeatedly came up as an important modulating factor. The idea of a 'trusting' relationship with a peer or mentor was key in several respects, and represented a range of interconnecting themes about how the writer perceived a support actor.

Sometimes trust had to do with privacy. W7 talked about how sharing her work with someone meant she's "going to show them a very deep inner part of [her] mind", and mentioned that there may be times when you don't want to share something "extremely sensitive" with a person yet. In this situation, a computer may be a useful intermediary step as a kind of anonymous support actor.

Relatedly, trust also had to do with vulnerability. Many writers talked about the emotional difficulty of getting feedback on their work, and that sometimes they were looking for validation and affirmation, while other times they wanted a more critical response. W10 says, "If it's a poem that is really close to my heart, sometimes if I get feedback on it, I don't even look at the feedback

for a couple of days until I'm ready to receive it." W14 talked about the fear of not being good enough, and noted that a computer seemed to not trigger this kind of vulnerability. Her comments about this articulate in general the kind of fears writers have when getting support:

> I think I probably wouldn't feel self conscious [with a computer]. On the basis that I wouldn't have an ongoing social relationship outside of the editing relationship. I think the basis of a lot of self consciousness is maybe the hope, or the anticipation, that we might have some kind of interpersonal relationship, even in passing as acquaintances.

At other times trust was about how well the support actor understood the *writer's* unique characteristics. Someone who had read their work before, who understood their writing 'crutches' and where they tend to be strong, would give better feedback than someone who didn't know anything about them. W15 talked about how he gave more weight to someone who had seen a lot of his work before, even if he didn't like their feedback or was skeptical for some reason.

Finally, trust often had to do with respect and admiration. W10 talked about sending her manuscript "to the editor in chief of that press, because he has worked on a ton of books." W15 talked about a "deference to expertise." When discussing trust, writers often referred to many of these different aspects coming together to create a trusting relationship.

6.2.4 Writer Values

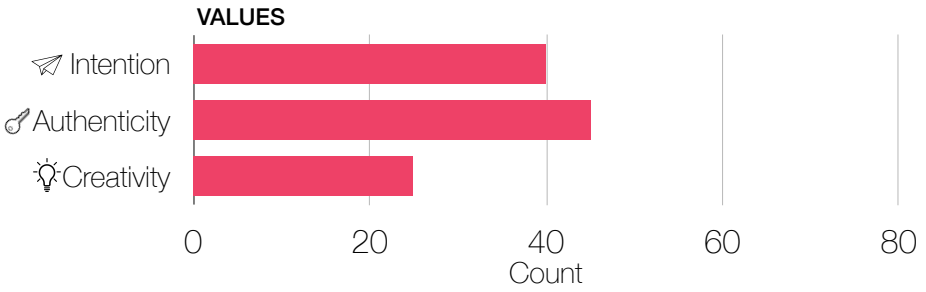


Figure 6.5: Prevalence of codes in data.

The writers varied on what they value when it comes to writing. For instance, some writers thought never having writer's block was key to their identity as a writer, while others were content

to experience this common plague. The idea that some parts of writing might be more important than others, and that writers disagree on what those parts are, is key to understanding when and why a writer turns to support, and what kind of support they'd like to have. There were three elements that writers continually brought up in the interviews, which we discuss here.

Intention The best support came from those who understood the writer's intention. Writer's talked about writing as being inherently subjective, and so their goals guided how a writer would (or would not) incorporate feedback or suggestions from peers and mentors. W4 described this as deciding "whether the feedback resonates with me or not". Others noted that suggestions were just suggestions, and they always had the authority to accept or reject them. Because there is no 'ground truth' for good writing, different writers are often trying to do different things. S19 used the metaphor of designing a roller coaster:

> Plato might not be as much fun as a rollick in a French novel, but they're all ultimately about asking a reader to engage and go on the ride with the creator. And what I think is super important and interesting is that there has been such meticulous decision making about how that ride is going to be shaped.

Due to this subjectivity, W1 talked about the importance of sometimes ignoring other people's opinions: "there is always a part where I stop and step back and ask if this is what I personally want to say. Divorced from 'is this what people want to hear'."

Support actors wouldn't always understand their intention. Writers talked about the difficulty of finding good readers who understood or respected their personal vision for a piece. Sometimes they'd have to discuss with a reader what they were trying to do in the writing, in order for the reader to know if they achieved that or not.

When it came to computers, this idea of sharing the intention was often a roadblock. W1 talked about how "it's not like the computer can understand what you want to say; they can only see what you have written." This is related to the idea of individuality in the support actor—some actors understand you better, or have a shared context—but relates specifically to the understanding of

your intention, which may be unique, personal, and hard to describe. S19 noted that a computer does not bring the same kind of intention that another person does. "Obviously, you have a reason why you are writing a story. And I think that is something SudoWrite can't can reproduce." W12 saw this as a benefit of working with a computer, noting that it's easier to "stay on track with a creative vision" if the computer doesn't have a creative vision of its own.

Intention is closely related to Chung et al.'s support relationship types [121], and theories of co-creativity more generally. Not a single writer in our study considered a human or computer support actor, even hypothetically, to be a leader (or even co-leader) in the writing process. Instead, support actors were always subordinate to their intention.

Authenticity Writers talked about authenticity, or their 'voice', as a concern when it came to incorporating the ideas or suggestions of others. Here, we describe four types of authenticity issues that came up in our interviews: 1) the *reader's* sense of authenticity, 2) the impact of viewing suggestions, 3) differing opinions on where authenticity lies, and 4) human v. computer authenticity issues. These four themes shed new light on the problem by addressing more specific concerns than authenticity generally.

First, some writers worried not about their own sense of authenticity, but the sense of their reader. Would the reader notice that the writing was not always in the writer's own voice? W11 said that readers were very perceptive, and that using a computational helper may be similar to writing in an unfamiliar genre: "if you tried to strain to a genre that you may be not experienced in or you're not that interested in, the audience might be able to suspect that." This, he said, was the same reason he also didn't believe in using ghost writers—he said the reader can tell when the author is not being authentic.[3]

Second, writers worried not just about ownership of words on the page, but how even viewing suggestions might derail them. W1 commented that, "once something is on the page it becomes just a bit harder to imagine what else it can be." W12 noted that the longer she looks at something,

[3]Unknowingly responding to this, S13, a professional genre fiction writer, noted that her beta readers never noticed when she started using SudoWrite. Some beta readers would even comment on phrases they particularly liked, and these phrases would be phrases written by SudoWrite. She was shocked.

the more she likes it, worrying that in moments of difficulty she would come to unconsciously or unintentionally prefer the computer suggestion.

Third, writers had different ideas about which parts of the writing process were most central to their feeling of authenticity. W10 talked about how "the ending is such a key piece of a poem that to have a computer do it would feel like cheating", while others were eager to get help with endings. S20 considered coming up with the storyline to be "a very human process", but was happy to use computers for overcoming writer's block. Conversely, W5 said overcoming writer's block and writing every day "makes me feel more like a writer" and would never ask anyone or thing to help him with that.

Fourth, some writers had existing close relationships with other writers, and these writers often left strong marks on their writing. In describing what was different between her brother influencing her writing and a computer, W8 said, "it somehow feels like there's more of me in it, maybe because we have a relationship with each other." W5 discussed how he gives trust to his fiance in a multitude of ways, and he can't imagine giving that same trust to a computer.

Creativity The question of if and how computers can be creative has been explored by researchers [9, 127] and critics [128] alike. In our interviews, we found writers bringing up this question when considering when they would feel comfortable with a computer influencing their writing. Connected to the idea of authenticity, writers considered if a computer could be creative, and how creative computational influence might impact the authenticity of their work.

Writers disagreed on if the computer could be creative. W1 proposed that what computers produce comes out of what they have seen (i.e. training data) and thus computers "help us understand or generalize what is a well-trodden path". W11 thought a computer would be better at historical novels than science fiction ones for this same reason; he trusted the computer to understand the past but not predict the future. W6 said, "I sort of hold the line in believing that there is something irreducibly human" that a computer could never replicate. Similarly W10 said, "I think very highly of all my friends' creative brains. But I don't think that way about a computer."

Others, especially SudoWrite users, were more positive about the creative potential of a computer. W10 even said she may feel bad compared to the computer's abilities, saying "I feel like it would bruise my ego if I couldn't figure out how to end one of my poems, but a computer came up with this really great idea." S19 related SudoWrite to role playing games that used dice rolls to trigger an idea for world building, saying SudoWrite is essentially a more evocative version of that. S20 discussed a moment when she had SudoWrite complete a scene she had started writing, and it "introduced a completely new character, and gave him two stanzas of a song that he was singing... I was floored." Even still, S20 said her creativity is her "humanity" and was not worried about a computer replacing that. She thought writers should be open to new technology, and make use of whatever was available to them.

W4, a TV comedy writer, had a unique perspective on the creative potential of computers. She discussed the difficulty of writing humorous scenes when people have seen so much TV already. She imagined using a computer to push herself and her writing further, saying, "I might take a first stab at a scene, then I would give the computer the first bit and be like 'you write it'. And then if I wrote the same thing as the computer, I'd know I have to do better." She explained further, "I feel like comedy rests on surprise. So I wouldn't trust a computer to do that. Or if the computer did do that, then there has to be something better."

Overall, many writers thought computers would struggle to be creative, and most held the perspective that, even if computers were to achieve the creativity they imagined, there would always be something irreducibly human that would distinguish their work from that of a computer.

6.3 Discussion

The social-technical gap describes the space between human-human dynamics (highly flexible, nuanced, and contextualized) and human-computer dynamics (rigid, brittle, and unchanging) [129]. In this work, we sought to answer the question *When and why might a creative writer turn to a computer versus a peer or mentor to provide support?* and outlined the rich and sophisticated social dynamics between writers and support actors. In this section, we discuss where computers

might excel, and where they may fall short, based on our understanding of the social-technical gap of computational writing support systems.

Writers don't worry about a computer's feelings, letting computers actually get closer to the writer's process. Writers worried about the relationships they with the people who gave them help. Was the writer asking for help too often? Was the writer's question too obvious, or their concern too insecure? Would the person helping them think the writing was bad? Comparatively, writers were not concerned about maintaining a good relationship with a computer program, and weren't worried about judged. Computers may best serve writers in tasks that feel too mundane, too frequent, or too worrisome to turn to a peer or mentor. This needn't just be spell-check; many writers were interested in programs that could suggest potential endings, point out possible problems, or provide new ideas. But likely computers will be considered a kind of pre-cursor to human support, perhaps even providing a very private exchange. As W14 said, "[a] computer program almost feels like me being in conversation with myself".

The personal relationships writers have with peers will be difficult, if not impossible, to mimic with computers. Writer's developed relationships with those who gave them help, whether it be the long-term relationship of a sibling, the blooming relationship of a new friend, or the structured relationship of a hired editor. While SudoWrite users did develop an *understanding* of the computational support, they expressly stated it was not personal. Computers will likely struggle to provide the individuality that personal relationships entail, like that of W10's brother who was also an Indian living in America. In the context of language models, this idea has nuance—many researchers think of pre-trained language models as general purpose[4] while in fact they do represent a particular and unique viewpoint based on their training data. Since most training data is scraped from the web, 'general purpose' models typically reflect white, western, and male perspectives, as these are the highest contributors to textual content on the web. Making explicit these assumptions, as well as highlighting when a model is trained to produce a different perspective (e.g. a model trained on woman-written science fiction, or on contemporary Latin American poetry), will be an

[4]GPT-3 [15] stands for General Purpose Transformer, v3.

important part of users developing strong mental models of a system. But it won't necessarily result in writers seeing these models as trusted readers. Computational models, according to our interviewees, are inherently reductive. A writer may get help from a friend who is queer, and this friend may be trusted to read a story from a queer perspective. But that friend doesn't represent all queer people, and writers likely are not interested in computers claiming to represent certain identity groups. It is in this way that computers will struggle to provide writers with personal relationships.

Computers may get better at understanding a writer's intention, but more research is required. A big roadblock for any kind of support was if the support actor understood the writer's intention. Feedback often failed to address what a writer needed because the support actor didn't understand what the writer was trying to do. Similarly, writers were skeptical a computer could do this when even peers sometimes failed. However, we see a future where computers can help writers articulate or understand their own intention. While intention may not always be found in what has already been written, computers may be able to guess at the intention based what has been provided, and provide a mirror for the work that the writer may find useful. But little work today explicitly models a writer's intention, or attempts to evaluate how well a system was able to understand or align itself with an intention. More research is required to test this idea.

6.4 Future Work

Our interviews pointed out a number of fruitful areas of research for system designers:

6.4.1 Feedback with specificity.

When writers talked about how other people influenced their writing, they almost always talked about feedback; no participant discussed other people writing for them. Despite this, most writing support tools studied in HCI generate text for a writer to use in their writing project, rather than support the reviewing process in some way. While there are exceptions [98, 130], they are in the minority [131]. Writers said the most helpful feedback was specific and they could ask questions

about it. For instance, indicating a paragraph might be confusing would be less useful than indicating a particular sentence referenced an idea the reader may not have heard of before. Writers wanted to be able to ask 'why' questions about feedback, such that they could drill into underlying issues. If a support actor says a poem is too abstract, a writer should be able to ask why and have the actor explain, perhaps, that the poem contains very few images of the natural world.[5]

6.4.2 Explainable feedback.

The desire for feedback that is specific and can be questioned points to a potential intersection of writing support tools and explainable AI (XAI). XAI has exploded as an important field of study for any system making use of neural networks, as neural networks typically lack clear reasons for their outputs [132]. Writing feedback may prove, in addition to being an important area of study for writing tools, a useful testbed for natural language XAI systems. Writing is a complicated activity and XAI systems would have to reason about the text in a sophisticated way.

6.5 Limitations

Though we tried to recruit widely, our participant population limits drawing extensive conclusions. For instance, only 6 of our 20 participants had an Master's of Fine Art in Creative Writing; most had only informal writing education. We also didn't collect publication information, but based on the interviews it seemed several participants were currently attempting to publish their first book. This suggests our population may be skewed slightly toward the amateur, and future studies may benefit from sampling writers with a longer publication history. And our SudoWrite users, while perhaps representative of all SudoWrite users, were exclusively fiction writers; mostly fantasy and science fiction novelists. Their experiences with SudoWrite likely don't represent the experiences of people writing poetry, nonfiction essays, memoirs, or novels of different genres. Researchers may also be interested in writing populations that may have different norms, such as fan fiction writers.

[5]We could imagine further follow-up questions, such as why should a poem contain images of the natural world?

Chapter 7: Conclusion and Future Work

In this thesis I asked, *How can generative systems support writers in constrained, creative writing tasks?* I demonstrated that by focusing on more narrow writerly goals, like writing a metaphor or explaining a technical concept, generative systems can provide coherent suggestions that writers use for inspiration, translation, and perspective. Moreover, through interviews with writers, I found that the social dynamics that underlie writer's desires for their writing, their perception of support actors, and their values in writerly interactions modulate their response to generative systems.

This thesis pushed the limits of how natural language generation can support writers, providing quantitative and qualitative evidence of writers working on self-motivated, 'ecologically valid' writing tasks with the aid of computer-generated text. This success opens up new areas of research. I presents three lines of future work. First, I present new and challenging writing support tasks that future work could address. Second, I outline open questions about how generative systems are used by writers. Third, I propose work on better shared methodologies to make the evaluation of writing support tools more robust.

7.1 New and Challenging Writing Support Tasks

7.1.1 Explainable Feedback

In Chapter 3 I showed that both the planning and reviewing cognitive processes in writing lack work; this thesis focused on planning and idea generation, but reviewing remains a relatively open area. Similarly, in Chapter 6 I reported on findings that many writers were interested in feedback with high specificity and that they could ask questions of. Future work should investigate the best ways to provide writers with feedback.

Early work on feedback for academic writing used templated questions that probed the rela-

tion between citations and the current work [cite]. (Also work from ETS?) While successful, the templated approach limits the diversity of the feedback, and doesn't allow writers to pose questions about the feedback. More recently, Arnold et al. [cite] have proposed posing questions to the reader to support writing, but these questions are intended to be used while writing, rather than while reviewing. Thus this remains an open problem.

Feedback should be specific, referring to particular instances in the text. Users should also be able to query the feedback, essentially asking 'why' questions to better understand what the feedback really represents. This represents a challenging new task for generative support systems, where instead of generating ideas the system generates coherent, cogent feedback for the writer.

7.1.2 Reader Perspectives

In Chapter 5 I found that incorrect text segments generated by a system could be used as a representation of a potential reader. For instance, if a system associates 'deprivation index' with obesity rather than economic status, a writer might find this incorrect association gives them insight into the associations their reader might be bringing. This suggests a unique use case of the understanding of bias in language models, and work could investigate how to best leverage the 'perspective' of a language model to provide a writer with approximations of their reader.

Dang et al. [cite] have begun to explore this in their work on using automatic text summaries as a writing tool. In particular some writers would use the summaries as a reader's perspective, and if the summary missed the main point of a paragraph they would revise that paragraph. While this work is in the direction of providing reader perspectives, there is much more to consider. First of all, summarization is not the only way to provide writers with a reader's perspective; we could also consider X, Y, and Z. Second, Dang et al. don't consider the details of the language model used, or how different language models (or uses of the same language model) might result in different responses to the text. Finally, their provided only a case study evaluation of the system.

7.1.3 New Domains

law, journalism, scientific writing e.g. paper writing,

7.2 How Generative Systems Are Used

This thesis presented evaluations of writing support tools that focused on using self-motivated tasks where writers brought their own topic or intention to the writing task. This ensured the results reflected how a writer might use such a system, and tested the systems in a variety of contexts. However, these studies were laboratory studies, with less than 20 participants and lasting only an hour. My findings could be validated at larger and longer scales.

7.2.1 Do these results hold up at scale?

While I presented use cases of the systems in this thesis, I did not make claims about the relative importance or prevalence of these use cases. Larger scale studies could investigate populations of writers, and what kinds of use cases are most popular.

Large studies could also study correlations between writer's attitudes and their use cases. For instance, do writers who value authenticity more highly edit system suggestions more heavily?

7.2.2 How do writers develop mental models of AI systems?

People develop mental models of systems over time. Especially with neural network-based technologies, the abilities and limits of a system are difficult to define or explain, even for experts [cite]. Studies of writers using support tools over days, weeks, or months will be able to study how they develop a mental model of such a complex system. What kinds of misconceptions to they develop, if any? How does their usage change over time as they discover which features work best for their particular use case? If there is an attrition of users over time, can we understand what predicts such attrition?

7.2.3 What are the pedagogical implications of long-term usage?

e.g do writers become better at ideation on their own, and phase out the use of the system? or do they become reliant on it?

7.3 Shared Evaluation Methodologies

7.3.1 Validated Writing Support Survey

similar to creativity support index, but specific to writing and taking into account the social dynamics of seeking support

7.3.2 Meta-Review of Interaction Measures

Work on writing support tools have presented a variety of ways to measure interaction with their system. The field could benefit from a meta-review that collected all the different interaction measures and proposed a unified framework for the interaction evaluation.

References

[1] M. Roemmele, "NEURAL NETWORKS FOR NARRATIVE CONTINUATION," PhD thesis, University of Southern California, 2018.

[2] E. Clark, A. S. Ross, C. Tan, Y. Ji, and N. A. Smith, "Creative writing with a machine in the loop, Case studies on slogans and stories," in *23rd International Conference on Intelligent User Interfaces*, ser. IUI '18, Tokyo, Japan: ACM, Mar. 2018, pp. 329–340, ISBN: 978-1-4503-4945-1.

[3] A. Calderwood, V. Qiu, K. I. Gero, and L. B. Chilton, "How novelists use generative language models: An exploratory user study," in *Human-AI Co-Creation with Generative Models*, Tokyo Japan: ACM, Mar. 2018, ISBN: 978-1-4503-4945-1.

[4] L. Flower and J. R. Hayes, "A cognitive process theory of writing," *Coll. Compos. Commun.*, vol. 32, no. 4, p. 365, Dec. 1981.

[5] J. R. Hayes, "A new framework for understanding cognition and affect in writing," in *The Science of Writing: Theories, Methods, Individual Differences, and Applications*, Lawrence Erbaum Associates, 1996.

[6] J. Emig, "Writing as a mode of learning," *Coll. Compos. Commun.*, vol. 28, no. 2, p. 122, May 1977.

[7] M. Scardamalia and C. Bereiter, "Knowledge telling and knowledge transforming in written composition," in *Advances in applied psycholinguistics*, Cambridge University Press, 1987.

[8] M. Csikszentmihalyi, "Implications of a Systems Perspective for the Study of Creativity," in *Handbook of Creativity*, R. J. Sternberg, Ed., 1st ed., Cambridge University Press, Oct. 1998, pp. 313–336, ISBN: 978-0-521-57285-9 978-0-521-57604-8 978-0-511-80791-6.

[9] M. A. Boden, "Computer Models of Creativity," *AI Magazine*, p. 12, 2009.

[10] T. M. Amabile, "The social psychology of creativity: A componential conceptualization.," *J. Pers. Soc. Psychol.*, vol. 45, no. 2, pp. 357–376, Aug. 1983.

[11] M. Sharples, "An Account of Writing as Creative Design," in *The Science of Writing*, 1st ed., Routledge, 1996, p. 22.

[12] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, *Distributed Representations of Words and Phrases and their Compositionality*, Oct. 2013.

[13] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A Neural Probabilistic Language Model," *Journal of Machine Learning Research*, 2003.

[14] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," p. 24,

[15] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," *arXiv:2005.14165 [cs]*, Jul. 2020, arXiv: 2005.14165.

[16] D. Adiwardana, M.-T. Luong, D. R. So, J. Hall, N. Fiedel, R. Thoppilan, Z. Yang, A. Kulshreshtha, G. Nemade, Y. Lu, and Q. V. Le, "Towards a human-like open-domain chatbot," *arXiv:2001.09977 [cs, stat]*, Feb. 2020, arXiv: 2001.09977.

[17] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The curious case of neural text degeneration," *arXiv:1904.09751 [cs]*, Feb. 2020, arXiv: 1904.09751.

[18] D. Ippolito, R. Kriz, M. Kustikova, J. a. Sedoc, and C. Callison-Burch, "Comparison of diverse decoding methods from conditional language models," *arXiv:1906.06362 [cs]*, Jun. 2019, arXiv: 1906.06362.

[19] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, "On the dangers of stochastic parrots, Can language models be too big?" In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, Virtual Event Canada: ACM, Mar. 2021, pp. 610–623, ISBN: 978-1-4503-8309-7.

[20] K. Kuddus, *Artificial intelligence in language learning: Practices and prospects*, May 2022. arXiv: 2202.03286 [cs.CL].

[21] A. Fan, M. Lewis, and Y. Dauphin, "Hierarchical neural story generation," *arXiv:1805.04833 [cs]*, May 2018, arXiv: 1805.04833.

[22] C. Meister, T. Vieira, and R. Cotterell, "Best-first beam search," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 795–809, Dec. 2020, arXiv: 2010.02650.

[23] L. Reynolds and K. McDonell, "Prompt programming for large language models: Beyond the few-shot paradigm," *arXiv:2102.07350 [cs]*, Feb. 2021, arXiv: 2102.07350.

[24] X. L. Li and P. Liang, "Prefix-tuning: Optimizing continuous prompts for generation," *arXiv:2101.00190 [cs]*, Jan. 2021, arXiv: 2101.00190.

[25] X. Liu, Y. Zheng, Z. Du, M. Ding, Y. Qian, Z. Yang, and J. Tang, *GPT understands, too*, 2021. arXiv: 2103.10385 [cs.CL].

[26] F. J. Damerau, "A technique for computer detection and correction of spelling errors," *Commun. ACM*, vol. 7, no. 3, 171–176, 1964.

[27] J. L. Peterson, "Computer programs for detecting and correcting spelling errors," *Communications of the ACM*, vol. 23, no. 12, pp. 676–687, Dec. 1980.

[28] C. Leacock, M. Chodorow, M. Gamon, and J. Tetreault, "Automated grammatical error detection for language learners," *Synth. Lect. Hum. Lang. Technol.*, vol. 3, no. 1, pp. 1–134, Jan. 2010.

[29] T. Ge, F. Wei, and M. Zhou, "Fluency boost learning and inference for neural grammatical error correction," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 1055–1065.

[30] N. Garay-Vitoria and J. Abascal, "Text prediction systems: A survey," *Universal Access in the Information Society*, vol. 4, no. 3, pp. 188–203, Mar. 2006.

[31] P. Quinn and S. Zhai, "A cost-benefit study of text entry suggestion interaction," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ser. CHI '16, San Jose, California, USA: Association for Computing Machinery, 2016, 83–88, ISBN: 9781450333627.

[32] M. X. Chen, B. N. Lee, G. Bansal, Y. Cao, S. Zhang, J. Lu, J. Tsay, Y. Wang, A. M. Dai, Z. Chen, T. Sohn, and Y. Wu, "Gmail smart compose, Real-time assisted writing," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Anchorage AK USA: ACM, Jul. 2019, pp. 2287–2295, ISBN: 978-1-4503-6201-6.

[33] A. Kannan, K. Kurach, S. Ravi, T. Kaufmann, A. Tomkins, B. Miklos, G. Corrado, L. Lukacs, M. Ganea, P. Young, and V. Ramavajjala, "Smart reply, Automated response suggestion for email," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco California USA: ACM, Aug. 2016, pp. 955–964, ISBN: 978-1-4503-4232-2.

[34] K. C. Arnold, K. Chauncey, and K. Z. Gajos, "Sentiment bias in predictive text recommendations results in biased writing," *Graphics Interface Conference*, p. 8, 2018.

[35] J. S. Hui, D. Gergle, and E. M. Gerber, "IntroAssist: A tool to support writing introductory help requests," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, ser. CHI '18. Montreal QC, Canada: Association for Computing Machinery, 2018, pp. 1–13, ISBN: 9781450356206.

[36] N. Maiden, K. Zachos, A. Brown, G. Brock, L. Nyre, A. Nygård Tonheim, D. Apsotolou, and J. Evans, "Making the news, Digital creativity support for journalists," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, ser. CHI '18, Montreal QC, Canada: ACM, Apr. 2018, 475:1–475:11, ISBN: 978-1-4503-5620-6.

[37] M. Roemmele and A. S. Gordon, "Automated assistance for creative writing with an RNN language model," in *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, ser. IUI '18 Companion, Tokyo, Japan: ACM, Mar. 2018, 21:1–21:2, ISBN: 978-1-4503-5571-1.

[38] M. S. Bernstein, G. Little, R. C. Miller, B. Hartmann, M. S. Ackerman, D. R. Karger, D. Crowell, and K. Panovich, "Soylent, A word processor with a crowd inside," in *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, ser. UIST '10, New York, New York, USA: ACM, Oct. 2010, pp. 313–322, ISBN: 978-1-4503-0271-5.

[39] "Heteroglossia: In-Situ Story Ideation with the Crowd," Honolulu HI USA, ISBN: 978-1-4503-6708-0.

[40] S. Andolina, H. Schneider, J. Chan, K. Klouche, G. Jacucci, and S. Dow, "Crowdboard: Augmenting in-person idea generation with real-time crowds," in *Proceedings of the 2017 ACM SIGCHI Conference on Creativity and Cognition*, New York, NY, USA: Association for Computing Machinery, 2017, ISBN: 9781450344036.

[41] J. Chan, S. Dang, and S. P. Dow, "Improving crowd innovation with expert facilitation," in *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work amp; Social Computing*, ser. CSCW '16, New York, NY, USA: Association for Computing Machinery, 2016, ISBN: 9781450335928.

[42] A. MacLean, R. Young, V. Bellotti, and T. Moran, "Questions, options, and criteria: Elements of design space analysis," *Hum-comput. Interact.*, vol. 6, no. 3, pp. 201–250, Sep. 1991.

[43] R. F. Woodbury and A. L. Burrow, "Whither design space?" *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 20, no. 2, pp. 63–82, Mar. 2006.

[44] K. Romer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wirel. Commun.*, vol. 11, no. 6, pp. 54–61, Dec. 2004.

[45] M. Roemmele and A. S. Gordon, "Automated assistance for creative writing with an RNN language model," in *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, ser. IUI '18 Companion, Tokyo, Japan: ACM, Mar. 2018, ISBN: 9781450355711.

[46] T. Wambsganss, C. Niklaus, M. Cetto, M. Sollner, S. Handschuh, and J. M. Leimeister, "AL: An adaptive learning support system for argumentation skills," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1–14, ISBN: 9781450367080.

[47] J. Frich, L. MacDonald Vermeulen, C. Remy, M. M. Biskjaer, and P. Dalsgaard, "Mapping the landscape of creativity support tools in HCI," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, Glasgow Scotland Uk: ACM, May 2019, pp. 1–18, ISBN: 978-1-4503-5970-2.

[48] F. Goncalves and P. Campos, "Understanding and evaluating the user interface design for creative writing," in *Proceedings of the European Conference on Cognitive Ergonomics 2017*, ser. ECCE 2017, Umea, Sweden: ACM, Sep. 2017, pp. 85–92, ISBN: 9781450352567.

[49] J. S. Olson, D. Wang, G. M. Olson, and J. Zhang, "How people write together now, Beginning the investigation with advanced undergraduates in a project course," *ACM Trans. Comput.-Hum. Interact.*, vol. 24, no. 1, pp. 1–40, Mar. 2017.

[50] E. P. P. Pe-Than, L. Dabbish, and J. D. Herbsleb, "Collaborative writing on GitHub, A case study of a book project," in *Companion of the 2018 ACM Conference on Computer Supported Cooperative Work and Social Computing*, ser. CSCW '18, Jersey City, NJ, USA: ACM, Oct. 2018, pp. 305–308, ISBN: 9781450360180.

[51] P. C. Hogan, "Literary style," in *Style in Narrative*, New York, NY, USA: Oxford University Press, Dec. 2020, pp. 23–72, ISBN: 9781450367080.

[52] K. I. Gero and L. B. Chilton, "How a stylistic, machine-generated thesaurus impacts a writer's process," in *Proceedings of the 2019 on Creativity and Cognition*, San Diego, CA, USA: ACM, Jun. 2019, pp. 597–603, ISBN: 9781450359177.

[53] E. Clark, A. S. Ross, C. Tan, Y. Ji, and N. A. Smith, "Creative writing with a machine in the loop, Case studies on slogans and stories," in *23rd International Conference on Intelligent User Interfaces*, ser. IUI '18, Tokyo, Japan: ACM, Mar. 2018, pp. 329–340, ISBN: 9781450349451.

[54] H. Osone, J.-L. Lu, and Y. Ochiai, "BunCho: Ai supported story co-creation via unsupervised multitask learning to increase writers' creativity in Japanese," in *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2021, ISBN: 9781450380959.

[55] K. I. Gero and L. B. Chilton, "Metaphoria, An algorithmic companion for metaphor creation," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, ser. CHI '19, Glasgow, Scotland Uk: ACM, May 2019, pp. 1–12, ISBN: 9781450359702.

[56] H. L. Han, M. A. Renom, W. E. Mackay, and M. Beaudouin-Lafon, "Textlets: Supporting constraints and consistency in text documents," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1–13, ISBN: 9781450367080.

[57] Z. Peng, Q. Guo, K. W. Tsang, and X. Ma, "Exploring the effects of technological writing assistance for support providers in online mental health community," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1–15, ISBN: 9781450367080.

[58] D. Schmidt, "Grading Tibetan children's literature, A test case using the nlp readability tool "dakje"," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 19, no. 6, pp. 1–19, Nov. 2020.

[59] X. T. Toyozaki and K. Watanabe, "AmbientLetter: Letter presentation method for discreet notification of unknown spelling when handwriting," in *The 31st Annual ACM Symposium on User Interface Software and Technology Adjunct Proceedings*, ser. UIST '18 Adjunct, Berlin, Germany: ACM, Oct. 2018, pp. 36–38, ISBN: 9781450359498.

[60] K. Watanabe, Y. Matsubayashi, K. Inui, T. Nakano, S. Fukayama, and M. Goto, "LyriSys, An interactive support system for writing lyrics based on topic transition," in *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, ser. IUI '17, Limassol, Cyprus: ACM, Mar. 2017, pp. 559–563, ISBN: 9781450343480.

[61] S. T. Iqbal, J. Teevan, D. Liebling, and A. L. Thompson, "Multitasking with play write, a mobile microproductivity writing tool," in *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '18, Berlin, Germany: ACM, Oct. 2018, pp. 411–422, ISBN: 9781450359481.

[62] J. Garbe, M. Kreminski, B. Samuel, N. Wardrip-Fruin, and M. Mateas, "StoryAssembler, An engine for generating dynamic choice-driven narratives," in *Proceedings of the 14th International Conference on the Foundations of Digital Games*, ser. FDG '19, San Luis Obispo, California, USA: ACM, Aug. 2019, ISBN: 9781450372176.

[63] M. Roemmele and A. S. Gordon, "Automated assistance for creative writing with an RNN language model," in *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, ser. IUI '18 Companion, Tokyo, Japan: ACM, Mar. 2018, ISBN: 9781450355711.

[64] X. Liu, A. Xu, Z. Liu, Y. Guo, and R. Akkiraju, "Cognitive learning, How to become william shakespeare," in *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, ser. CHI EA '19, Glasgow, Scotland Uk: ACM, May 2019, pp. 1–6, ISBN: 9781450359719.

[65] E. Cherry and C. Latulipe, "Quantifying the creativity support of digital tools through the creativity support index," *ACM Trans. Comput.-Hum. Interact.*, vol. 21, no. 4, pp. 1–25, Aug. 2014.

[66] S. G. Hart and L. E. Staveland, "Development of NASA}-{TLX (Task load index): Results of empirical and theoretical research," in *Advances in Psychology*, vol. 52, Elsevier, 1988, pp. 139–183.

[67] V. Venkatesh and F. D. Davis, "A theoretical extension of the technology acceptance model: Four longitudinal field studies," *Manage. Sci.*, vol. 46, no. 2, pp. 186–204, Feb. 2000.

[68] E. Manjavacas, F. Karsdorp, B. Burtenshaw, and M. Kestemont, "Synthetic literature: Writing science fiction in a co-creative process," in *Proceedings of the Workshop on Computational Creativity in Natural Language Generation (CC-NLG 2017)*, Association for Computational Linguistics, 2017, pp. 29–37.

[69] R. Sloan, *Writing with the machine*, <https: //www.robinsloan.com/notes/writing-with-the-machine/>, Accessed: 2018-09-19, 2016.

[70] J. Jacobs, J. Brandt, R. Mech, and M. Resnick, "Extending manual drawing practices with artist-centric programming tools," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, ser. CHI '18, Montreal QC, Canada: ACM, Apr. 2018, 590:1–590:13, ISBN: 978-1-4503-5620-6.

[71] G. Fauconnier and M. Turner, *The way we think: Conceptual blending and the mind's hidden complexities*. Basic Books, 2008.

[72] P. J. Silvia and R. E. Beaty, "Making creative metaphors: The importance of fluid intelligence for creative thought," *Intelligence*, vol. 40, no. 4, pp. 343–351, Jul. 2012.

[73] T. Veale, E. Shutova, and B. B. Klebanov, "Metaphor: A computational perspective," *Synth. Lect. Hum. Lang. Technol.*, vol. 9, no. 1, pp. 1–160, Feb. 2016.

[74] T. Veale, *Thesaurus rex*, <http://ngrams.ucd.ie/therex3/>, Accessed: 2018-09-19.

[75] T. Veale and Y. Hao, "Comprehending and generating apt metaphors: A web-driven, case-based approach to figurative language," in *AAAI*, vol. 2007, 2007, pp. 1471–1476.

[76] A. Gagliano, E. Paul, K. Booten, and M. A. Hearst, "Intersecting word vectors to take figurative language to new heights," in *Proceedings of the Fifth Workshop on Computational Linguistics for Literature*, Association for Computational Linguistics, 2016, pp. 20–31.

[77] T. Veale and G. Li, "Distributed divergent creativity: Computational creative agents at web scale," *Cogn. Comput.*, vol. 8, no. 2, pp. 175–186, May 2015.

[78] T. Veale, "Less rhyme, more reason: Knowledge-based poetry generation with feeling, insight and wit.," in *ICCC*, 2013, pp. 152–159.

[79] D. Gentner, "Structure-mapping: A theoretical framework for analogy*," *Cognitive Sci.*, vol. 7, no. 2, pp. 155–170, Apr. 1983.

[80] K. Gilon, J. Chan, F. Y. Ng, H. Liifshitz-Assaf, A. Kittur, and D. Shahaf, "Analogy mining for specific design needs," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, ser. CHI '18, Montreal QC, Canada: ACM, Apr. 2018, 121:1–121:11, ISBN: 978-1-4503-5620-6.

[81] H Liu and P Singh, "ConceptNet — a practical commonsense reasoning tool-kit," *Bt Technol. J.*, vol. 22, no. 4, pp. 211–226, Oct. 2004.

[82] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, 2014, pp. 1532–1543.

[83] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Understanding Morphology*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Routledge, Oct. 2013, ch. Words and phrases, pp. 205–226, ISBN: 9780203776506.

[84] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, "From word embeddings to document distances," in *International Conference on Machine Learning*, 2015, pp. 957–966.

[85] A. B. Warriner, V. Kuperman, and M. Brysbaert, "Norms of valence, arousal, and dominance for 13,915 English lemmas," *Behav. Res. Methods*, vol. 45, no. 4, pp. 1191–1207, Feb. 2013.

[86] O. Levy and Y. Goldberg, "Dependency-based word embeddings," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, vol. 2, Association for Computational Linguistics, 2014, pp. 302–308.

[87] T. Linzen, "Issues in evaluating semantic spaces using word analogies," *CoRR*, vol. abs/1606.07736, 2016. arXiv: 1606.07736.

[88] K. Compton, A. Smith, and M. Mateas, "Anza island, Novel gameplay using asp," in *Proceedings of the The third workshop on Procedural Content Generation in Games - PCG'12*, ACM Press, 2012, pp. 228–235.

[89] A. C. Breu, "From tweetstorm to tweetorials: Threaded tweets as a tool for medical education and knowledge dissemination," *Semin. Nephrol.*, vol. 40, no. 3, pp. 273–278, May 2020.

[90] A. Soragni and A. Maitra, "Of scientists and tweets," *Nat. Rev. Cancer*, vol. 19, no. 9, pp. 479–480, Jun. 2019.

[91] A. Shelby and K. Ernst, "Story and science, How providers and parents can utilize storytelling to combat anti-vaccine misinformation," *Human Vaccines & Immunotherapeutics*, vol. 9, no. 8, pp. 1795–1801, Aug. 2013.

[92] K.-C. Yang, F. Pierri, P.-M. Hui, D. Axelrod, C. Torres-Lugo, J. Bryden, and F. Menczer, "The {covid}-19 infodemic: Twitter versus facebook," *Big Data & Society*, vol. 8, no. 1, p. 205 395 172 110 138, Jan. 2021, arXiv: 2012.09353.

[93] P. S. Hart and E. C. Nisbet, "Boomerang effects in science communication, How motivated reasoning and identity cues amplify opinion polarization about climate mitigation policies," *Commun. Res.*, vol. 39, no. 6, pp. 701–723, Aug. 2011.

[94] S. A. Gilbert, ""I run the world's largest historical outreach project and it's on a cesspool of a website." moderating a public scholarship site on reddit: A case study of r/AskHistorians," *Proceedings of the ACM on Human-Computer Interaction*, vol. 4, no. CSCW1, pp. 1–27, May 2020.

[95] D. J. Welbourne and W. J. Grant, "Science communication on YouTube: Factors that affect channel and video popularity," *Public Underst. Sci.*, vol. 25, no. 6, pp. 706–718, Feb. 2015.

[96] M. Brüggemann, I. Lörcher, and S. Walter, "Post-normal science communication: Exploring the blurring boundaries of science and journalism," *Journal of Science Communication*, vol. 19, no. 03, A02, Jun. 2020.

[97] K. I. Gero and L. B. Chilton, "Metaphoria, An algorithmic companion for metaphor creation," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, Glasgow Scotland Uk: ACM, May 2019, pp. 1–12, ISBN: 978-1-4503-5970-2.

[98] Z. Peng, Q. Guo, K. W. Tsang, and X. Ma, "Exploring the effects of technological writing assistance for support providers in online mental health community," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, Honolulu HI USA: ACM, Apr. 2020, pp. 1–15, ISBN: 978-1-4503-6708-0.

[99] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. Rush, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Online: Association for Computational Linguistics, 2020, pp. 38–45.

[100] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter," *arXiv:1910.01108 [cs]*, Feb. 2020, arXiv: 1910.01108.

[101] W.-J. Ko, G. Durrett, and J. J. Li, "Domain agnostic real-valued specificity prediction," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 6610–6617, Jul. 2019.

[102] R. Zhang, J. Guo, Y. Fan, Y. Lan, J. Xu, and X. Cheng, "Learning to control the specificity in neural response generation," p. 10,

[103] K. I. Gero, C. Kedzie, S. Petridis, and L. Chilton, "Lightweight Decoding Strategies for Increasing Specificity," *arXiv:2110.11850 [cs]*, Oct. 2021.

[104] A. Vijayakumar, M. Cogswell, R. Selvaraju, Q. Sun, S. Lee, D. Crandall, and D. Batra, "Diverse beam search for improved description of complex scenes," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, Apr. 2018, arXiv: 1610.02424.

[105] A. C. Graesser, M. Singer, and T. Trabasso, "Constructing inferences during narrative text comprehension.," *Psychol. Rev.*, vol. 101, no. 3, pp. 371–395, 1994.

[106] B. J. F. Meyer and M. N. Ray, "Structure strategy interventions: Increasing reading comprehension of expository text," *International Electronic Journal of Elementary Education*, vol. 4, no. 1, pp. 127–152, Aug. 2017.

[107] T. Wu, M. Terry, and C. J. Cai, "AI chains: Transparent and controllable human-{ai} interaction by chaining large language model prompts," *arXiv:2110.01691 [cs]*, Oct. 2021, arXiv: 2110.01691.

[108] X. Li, A. Taheri, L. Tu, and K. Gimpel, "Commonsense knowledge base completion," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany: Association for Computational Linguistics, 2016, pp. 1445–1455.

[109] N. Reimers and I. Gurevych, "Sentence-{bert:} Sentence embeddings using Siamese {bert}-networks," *arXiv:1908.10084 [cs]*, Aug. 2019, arXiv: 1908.10084.

[110] E. L. Howell, J. Nepper, D. Brossard, M. A. Xenos, and D. A. Scheufele, "Engagement present and future: Graduate student and faculty perceptions of social media and the role of the public in science engagement," *PLoS One*, vol. 14, no. 5, Z. Master, Ed., e0216274, May 2019.

[111] E. Cherry and C. Latulipe, "Quantifying the creativity support of digital tools through the creativity support index," *ACM Trans. Comput.-Hum. Interact.*, vol. 21, no. 4, pp. 1–25, Aug. 2014.

[112] V. Braun and V. Clarke, "Thematic analysis.," in *APA handbook of research methods in psychology, Vol 2: Research designs: Quantitative, qualitative, neuropsychological, and biological.* H. Cooper, P. M. Camic, D. L. Long, A. T. Panter, D. Rindskopf, and K. J.

Sher, Eds., Washington: American Psychological Association, 2012, pp. 57–71, ISBN: 978-1-4338-1005-3.

[113] Y. Dou, M. Forbes, R. Koncel-Kedziorski, N. A. Smith, and Y. Choi, "Scarecrow: A framework for scrutinizing machine text," *arXiv:2107.01294 [cs]*, Jul. 2021, arXiv: 2107.01294.

[114] H. Shakeri, C. Neustaedter, and S. DiPaola, "SAGA: Collaborative storytelling with {gpt}-3," in *Companion Publication of the 2021 Conference on Computer Supported Cooperative Work and Social Computing*. New York, NY, USA: Association for Computing Machinery, 2021, pp. 163–166, ISBN: 9781450384797.

[115] A. Coenen, L. Davis, D. Ippolito, E. Reif, and A. Yuan, "Wordcraft: A human-{ai} collaborative editor for story writing," *arXiv:2107.07430 [cs]*, Jul. 2021, arXiv: 2107.07430.

[116] E. Clark, A. S. Ross, C. Tan, Y. Ji, and N. A. Smith, "Creative writing with a machine in the loop, Case studies on slogans and stories," in *23rd International Conference on Intelligent User Interfaces*, Tokyo Japan: ACM, Mar. 2018, pp. 329–340, ISBN: 978-1-4503-4945-1.

[117] C. Oh, J. Song, J. Choi, S. Kim, S. Lee, and B. Suh, "I lead, you help but only with enough details, Understanding user experience of co-creation with artificial intelligence," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, Montreal QC Canada: ACM, Apr. 2018, pp. 1–13, ISBN: 978-1-4503-5620-6.

[118] N Dehouche, "Plagiarism in the age of massive generative pre-trained transformers {(gpt}-3)," *Ethics in Science and Environmental Politics*, vol. 21, pp. 17–23, Mar. 2021.

[119] R. Sadeghi, "The attitude of scholars has not changed towards plagiarism since the medieval period: Definition of plagiarism according to shams-e-qays, thirteenth-century Persian literary scientist," *Research Ethics*, vol. 15, no. 2, pp. 1–3, May 2016.

[120] O. Schmitt and D. Buschek, "CharacterChat: Supporting the creation of fictional characters through conversation and progressive manifestation with a chatbot," in *Creativity and Cognition*, Virtual Event Italy: ACM, Jun. 2021, pp. 1–10, ISBN: 978-1-4503-8376-9.

[121] J. J. Y. Chung, S. He, and E. Adar, "Artist Support Networks: Implications for Future Creativity Support Tools," in *Designing Interactive Systems Conference*, Virtual Event Australia: ACM, Jun. 2022, pp. 232–246, ISBN: 978-1-4503-9358-4.

[122] K. Booten and K. I. Gero, "Poetry Machines: Eliciting Designs for Interactive Writing Tools from Poets," in *Creativity and Cognition*, Virtual Event Italy: ACM, Jun. 2021, pp. 1–5, ISBN: 978-1-4503-8376-9.

[123] R. J. So and G. Wezerek, "Just how white is the book industry?" *The New York Times*, Dec. 2020.

[124] R. J. So and A. Piper, "How has the mfa changed the contemporary novel?" *The Atlantic*, Mar. 2016.

[125] M. Q. Patton, *Qualitative Research and Evaluation Methods*, 3rd ed. Sage Publications, 2002.

[126] D. R. Thomas, "A General Inductive Approach for Analyzing Qualitative Evaluation Data," *American Journal of Evaluation*, vol. 27, no. 2, pp. 237–246, Jun. 2006.

[127] M. L. Maher, "Computational and Collective Creativity: Who's Being Creative?," p. 5, 2012.

[128] R. Millière, "Ai art is challenging the boundaries of curation," *Wired*, Jul. 2022.

[129] M. S. Ackerman, "The Intellectual Challenge of CSCW: The Gap Between Social Requirements and Technical Feasibility," *Human–Computer Interaction*, vol. 15, no. 2-3, pp. 179–203, Sep. 2000.

[130] M. Liu, R. A. Calvo, and V. Rus, "Automatic generation and ranking of questions for critical review," p. 15, 2020.

[131] K. Gero, A. Calderwood, C. Li, and L. Chilton, "A Design Space for Writing Support Tools Using a Cognitive Process Model of Writing," in *Proceedings of the First Workshop on Intelligent and Interactive Writing Assistants (In2Writing 2022)*, Dublin, Ireland: Association for Computational Linguistics, 2022, pp. 11–24.

[132] F. Xu, H. Uszkoreit, Y. Du, W. Fan, D. Zhao, and J. Zhu, "Explainable ai: A brief survey on history, research areas, approaches and challenges," in *Natural Language Processing and Chinese Computing*, J. Tang, M.-Y. Kan, D. Zhao, S. Li, and H. Zan, Eds., Cham: Springer International Publishing, 2019, pp. 563–574, ISBN: 978-3-030-32236-6.

[133] F. Gonçalves, A. Caraban, E. Karapanos, and P. Campos, "What shall i write next? Subliminal and supraliminal priming as triggers for creative writing," in *Proceedings of the European Conference on Cognitive Ergonomics 2017*, ser. ECCE 2017, Umea, Sweden: ACM, Sep. 2017, pp. 77–84, ISBN: 9781450352567.

[134] A. Guarneri, L. A. Ripamonti, F. Tissoni, M. Trubian, D. Maggiorini, and D. Gadia, "Ghost, A ghost story-writer," in *Proceedings of the 12th Biannual Conference on Italian SIGCHI Chapter*, ser. CHItaly '17, Cagliari, Italy: ACM, Sep. 2017, ISBN: 9781450352376.

[135] S. Türkay, D. Seaton, and A. M. Ang, "Itero, A revision history analytics tool for exploring writing behavior and reflection," in *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, ser. CHI EA '18, Montreal QC, Canada: ACM, Apr. 2018, pp. 1–6, ISBN: 9781450356213.

[136] L. Wang, X. Fan, F. Tian, L. Deng, S. Ma, J. Huang, and H. Wang, "mirrorU, Scaffolding emotional reflection via in-situ assessment and interactive feedback," in *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, ser. CHI EA '18, Montreal QC, Canada: ACM, Apr. 2018, pp. 1–6, ISBN: 9781450356213.

[137] E. LaBouve, E. Miller, and F. Khosmood, "Enhancing story generation with the semantic web," in *Proceedings of the 14th International Conference on the Foundations of Digital Games*, ser. FDG '19, San Luis Obispo, California, USA: ACM, Aug. 2019, ISBN: 9781450372176.

[138] S. Wu, L. Reynolds, X. Li, and F. Guzman, "Design and evaluation of a social media writing support tool for people with dyslexia," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, ser. CHI '19, Glasgow, Scotland Uk: ACM, May 2019, pp. 1–14, ISBN: 9781450359702.

[139] O. Resch and A. Yankova, "Open knowledge interface: A digital assistant to support students in writing academic assignments," in *Proceedings of the 1st ACM SIGSOFT International Workshop on Education through Advanced Software Engineering and Artificial Intelligence - EASEAI 2019*, ser. EASEAI 2019, Tallinn, Estonia: ACM Press, 2019, pp. 13–16, ISBN: 9781450368520.

[140] C.-Y. Huang, S.-H. Huang, and T.-H. K. Huang, "Heteroglossia: In-situ story ideation with the crowd," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1–12, ISBN: 9781450367080.

[141] E. P. P. Pe-Than, L. Dabbish, and J. Herbsleb, "Open collaborative writing, Investigation of the fork-and-pull model," *Proceedings of the ACM on Human-Computer Interaction*, vol. 5, no. CSCW1, pp. 1–33, Apr. 2021.

[142] S. Tian, A. X. Zhang, and D. Karger, "A system for interleaving discussion and summarization in online collaboration," *Proceedings of the ACM on Human-Computer Interaction*, vol. 4, no. CSCW3, pp. 1–27, Jan. 2021.

[143] N. S. Keskar, B. McCann, L. R. Varshney, C. Xiong, and R. Socher, "CTRL: A conditional transformer language model for controllable generation," *arXiv:1909.05858 [cs]*, Sep. 2019, arXiv: 1909.05858.

Appendix A: A Design Space for Writing Support Tools

A.1 Methodology

We chose our inclusion criteria subjectively, to focus on our particular interest in writing support tools and their relation to improvements in language technology. We do not intend to present this inclusion criteria as an objective definition of writing support tools. For instance, handwriting recognition may be considered a writing support tool in some contexts, but would not fit our purposes. Another small group of papers we rejected were papers that supported the collection or organization of data that would later be written about, such as a tool for quickly extracting sports-game highlights for sportswriters, and another that solicited reflections throughout the day to support memoir writing. Journalists and others may consider these writing tools, but we excluded them on the rationale that they were somewhat disconnected from the final text produced.

Table A.1 shows all annotations done for the papers selected.

Below we list all 30 papers selected for this review, with brief descriptions and ordered by the year they were published:

UI Design [48]: Presents a user study of four writing environments – Microsoft Word, Scrivener, OmniWriter and Ulysses. They found OmniWriter to be the most satisfying tool, and propose design guidelines for such tools, including full-screen mode for distraction-free writing.

LyriSys [60]: Reports on a lyric generation system, which generates full song lyrics according to strain and accent constraints, and provides plenty of user control including semantic topic transitions.

Writing Together [49]: Studies data traces of collaborative writing in student teams' use of Google Docs.

Liminal Triggers [133]: Investigates how subliminal triggering may help to relieve writer's

How support aligns with the cognitive process model	
part of writing process	plan / translate / review
level of constraint	1: low constraint (almost anything could be helpful) 3: medium constraint (constrained but with variety in "right" answers) 5: high constraint (support must be very specific, few "right" answers)
size of goal being support	word / sentence / paragraph / more than paragraph / writing experience
Matching creativity support tool review [47]	
complexity of tool	low: one or two features medium: multiple features, semi-complex system high: entire system or suite of tools
evaluation type	no evaluation / case study / qualitative / quantitative / mixed methods
number of participants	(numeric response)
evaluation criterion	(open response)
time spent writing with tool	(numeric response in minutes)
Quantifying type of research	
tool is exclusively about text	yes/no
tool is about collaborative writing	yes/no
tool is contribution	yes/no
technology tool uses	(open response)

Table A.1: List of all annotations done for the papers. Most annotations have options, while some are open response.

block.

GHOST [134]: Presents a tool to support non-writers creating stories for video games. The resulting tool, GHOST, is built into Unity and aids in the creation of plot roadmaps.

Writing with RNN [63]: Presents Creative Help, an interface that suggests new sentences in a story using an RNN language model. Study varies the degree of randomness.

MiL [53]: Presents and studies creative writing support tools: a next-sentence generator for story telling, and a slogan generator for writing slogans.

AmbientLetter [59]: Proposes a technique to support writing activity (via autocorrection and predictive conversion) in a confidential manner with a pen-based device.

Play Write [61]: Introduces a microproductivity tool that allows users to review and edit Word documents in small moments of spare time from their smartphone.

IntroAssist [**introassist**]: Presents a tool for supporting writing introductory help requests via email by providing checklists and examples.

Itero [135]: Presents a study on how integrating writing revision analytics and visualization into writing practices can impact writing self-efficacy.

Writing on Github [50]: Presents the preliminary findings of a mixed-methods, case study of collaboration practices in a GitHub book project.

MirrorU [136]: Presents a mobile system to support reflecting and writing about daily emotional experiences; provides assessment and feedback across level of detail, overall valence, and cognitive engagement.

Semantic Web [137]: Presents a mixed initiative tool for story generation, designed to take as input a story generating grammar in addition to generic keywords and uses the semantic web to contribute real-world details.

Shakespeare [64]: Presents a web application that helps with educating different writing styles through automatic style transfer (with deep learning), visual stylemotry analytics, and machine teaching (by picking out examples of a particular writing style). The authors propose a use case of this system with Shakspeare's writings.

Metaphoria [55]: Presents a tool that shows how words might be metaphorically related.

StoryAssembler [62]: Presents StoryAssembler, an open source generative narrative system that creates dynamic choice-driven narratives, and a case study.

SMWS [138]: This paper describes a tool built by the Facebook researchers to automatically 'translate' text written by people with dyslexia to non-dyslexic style writing. Having built the tool into the Facebook comment interfcae, they conduct a week long study to measure its efficacy.

Academic Writing [139]: Presents OKI, a chatbot tool that helps with project management, assistance in applying scientifc methods, and search in open access literature.

Style Thesaurus [52]: Presents a series of automatically generated thesauruses, using word embeddings trained on custom corpuses, which reflect the stylistic preferences of the corpus text.

AL [46]: This paper presents an NLP tool to aid student argumentative writing by providing automatic feedback on their argumentation structure.

Dakje [58]: Introduces a new readability tool alongside a specific use case, and demonstrates how it can help benefit literacy in the Tibetan languages. Users have instant access to statistics on the readability of their word choices so they can make edits for easy-to-read text.

Heteroglossia [140]: Presents a crowd-sourcing tool that allows writer to elicit story ideas based on a role-play strategy. The tool is developed as Google Doc add-on.

Textlets [56]: Introduces Textlets, interactive objects that reify text selections into persistent items, and show how Textlets can be used for selective search and replace, word count, and alternative wording.

MepsBot [57]: Presents in-situ writing assistance for people commenting in online mental health communities; compares support that assesses text versus recommends text.

Literary Style [51]: Develops a model of style by training a neural net, and present novel applications including an interactive text editor with real-time style feedback.

Fork-and-Pull [141]: Investigates the utility of the GitHub "fork and pull" workflow for writ-

ers through a mixed-methods case study of collaborative writing. They looked at two collaborative writing cases, the first to write a mathematics textbook on homotopy type theory, and the second a set of open source public policies.

IDS System [142]: Presents Wikum+, a website that allows you to create instances of interleaved discussion and summarization.

BunCho [54]: Presents a tool for generating titles and synopses from keywords. Additionally, an interactive story co-creation AI system is proposed. (Japanese language)

There was some ambiguity in the annotations. Some tools straddled multiple parts of the writing process, or the paper didn't frame the tool in a way that clearly defined the intention of the support. Systems that provided generated text were sometimes framed as providing ideas for the writer, and these labeled as supporting 'planning', whereas others that provided generated text were framed as actually writing, and these were labeled as supporting 'translating'. However, the distinction can be subtle, and sometimes, in a user study, participants used the tool in a different way than the designers intended. Some tools had a single main feature and many small 'satellite' features, making the level of complexity unclear. Our intention with these annotations is not to provide a perfectly objective representation but rather to understand the breadth and similarities within a field of study. When an annotator was unsure about an annotation, they consulted with the rest of the team.

Some papers presented or studied more than one tool; others presented more than one evaluation for a single tool. In the case of multiple tools, we give each tool its own nickname and consider them separate entities. In the case of multiple evaluations, we consider them separate entities only when analyzing evaluation methodologies. (Multiple tools evaluated together are considered a single entity when analyzing evaluation methodologies.)

Some papers studied existing commercial writing tools, and others presented novel tools developed by the researchers. The commercial writing tools studied tended to be word processors, like Microsoft Word or Google Docs. We include all of these in our analysis.

Appendix B: Sparks: Inspiration for Science Writing Using Language Models

B.1 System Design

B.1.1 Enumeration of Decoding Method

Let X be the prompt for the language model and Y be an output decoded from the language model given the prompt. Because we want multiple unique outputs from the same prompt, let Y^n be the n^{th} output decoded from the language model. Y^n is a sequence of tokens $(y_0^n, ..., y_i^n, ..., y_m^n)$; a partially decoded $Y_{0:i}^n$ would be $(y_0^n, ..., y_i^n)$.

At any given point in the generation process, let Z represent the set of all tokens in the vocabulary, such that any point we are selecting y_i from a probability distribution $P(Z|X + Y_{0:i-1})$.

Our decoding process is as follows:

1. let y_0^n be the n^{th} most likely token in $P(Z|X)$
2. while generating $Y_{1:m}^n$, select y_i^n from only the top 50 tokens in $P(Z|X + Y_{0:i-1})$
3. while generating $Y_{1:m}^n$, modify $P(Z|X + Y_{0:i-1})$ with the normalized inverse word frequency (only top 50 tokens from unmodified distribution will be considered, as per step 2)
4. perform beam search on the prefix $X + y_0^n$ with $k = 3$, selecting the top beam as the output

This decoding method was designed partially to be able to produce Y_n at any point, without having to generate $(Y^0, ..., Y^{n-1})$ or any $Y^{>n}$ outputs. This improves computation time, and while is common in sampling methods (where you can sample infinitely without requiring to know anything about previous samples) is not the case when using regular beam search to produce multiple outputs (where all n beams are generated at one time).

We also make use of two built-in huggingface functions for improving the quality of outputs.

First, a small repetition penalty, modeled off of [143][1], where we set the repetition penalty to 1.2. Second, a blacklist that includes words that commonly derailed the output, like the word 'figure' which often resulted in an output like 'See figure 2 for more details'. Our blacklisted words were:

> one, two, three, four, five, six, seven, eight, nine, ten, eleven, twelve, thirteen, fourteen, fifteen, sixteen, seventeen, eighteen, nineteen, 'twenty, tens, hundreds, thousands, millions, Figure, figure, Fig, fig, Chapter, chapter

This decoding process was developed iteratively while testing the system with a variety of pilot users and test topics. We would regularly generate the top 10 responses to topics across computer science and biology to look for common failure points, like redundant responses, generic responses, incoherent responses, and factually false responses.

B.2 Methodology: Study 1

B.2.1 Full List of Topics Studied

- **Biology**: endergonic reactions, genetic drift, decomposition, dynein, circadian rhythm, placebos, ethology, osmosis, reproductive biology, bioenergetics.
  Topics randomly sampled from `https://en.wikipedia.org/wiki/Glossary_of_biology`.
- **Environmental science**: biocapacity, resource productivity, forage, polypropylene, open-pit mining, soil conditioner, incineration, green marketing, coir, old growth forests.
  Topics randomly sampled from `https://en.wikipedia.org/wiki/Glossary_of_computer_science`.
- **Computer science**: source code, automata theory, computer security, control flow, boolean expressions, double-precision floating-point format, linear search, software development, hash functions, cyberbullying.

[1] And documented at `https://huggingface.co/transformers/v4.6.0/internal/generation_utils.html#transformers.RepetitionPenaltyLogitsProcessor`

Topics randomly sampled from https://en.wikipedia.org/wiki/Glossary_of_environmental_science.

B.3 Methodology: Study 2

B.3.1 Survey Questions

1. What year of your graduate program are you in?
2. What kind of graduate program are you in?
3. What discipline do you study?
4. How often do you write about technical topics for a general audience? e.g. blog posts, opinion articles, essays, etc.
5. How often do you post on Twitter about technical topics?

B.3.2 Interview Questions

■ Questions about the task:

1. Did you find any of the sparks helpful? If so, could you recall one spark that was helpful and explain in what way it helped? (Make sure to dig into how the spark related to what they eventually wrote. Ask them to point it out in what they wrote.)
2. How do you think the sparks differed from what you would find on Wikipedia? How about Google search, or some other resource you use often?
3. How did the existing prompts differ from your custom prompts?
4. Could you recall one spark that wasn't helpful, and explain why?
5. Were any of the sparks presented incorrect in some way? If so, what did you think of these?
6. What made you decide to stop generating sparks?
7. Did you have any concerns about ownership or agency?

■ Debriefing questions:

1. Is there anything you'd like to share that I didn't ask about?
2. Is there anything you'd like to know or ask me?

Appendix C: Social Dynamics of AI Support in Creative Writing

C.1 Methodology

C.1.1 SudoWrite Features

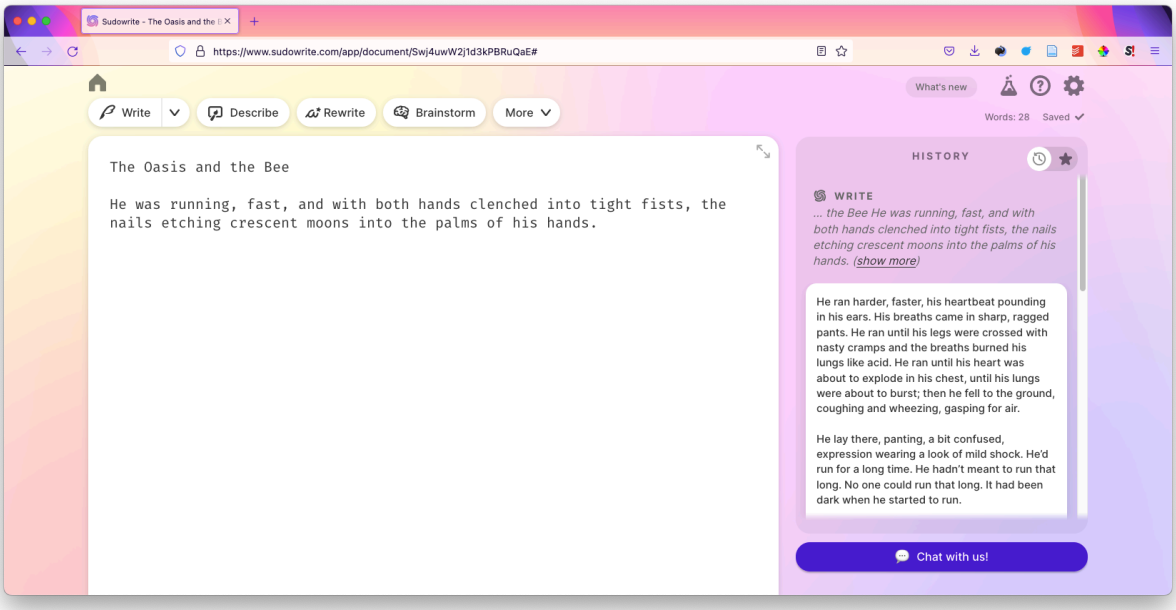


Figure C.1: 'Write' feature of SudoWrite, which continues from wherever the cursor is.

Figures C.1, C.2, and C.3 demonstrate some of the functionality of SudoWrite at the time of this study. Suggested text is shown on a panel on the right, where multiple options are available. For instance, when using the 'write' functionality, multiple different continuations are generated, and writers can easily paste them into the main text box. For the 'describe' functionality, different kinds of descriptions are generated, such as 'sight', 'smell', and 'metaphor'. The 'brainstorm' functionality has its own interface separate from the writing interface.

SudoWrite functionality has changed over time. Some users talked about the 'wormhole' func-

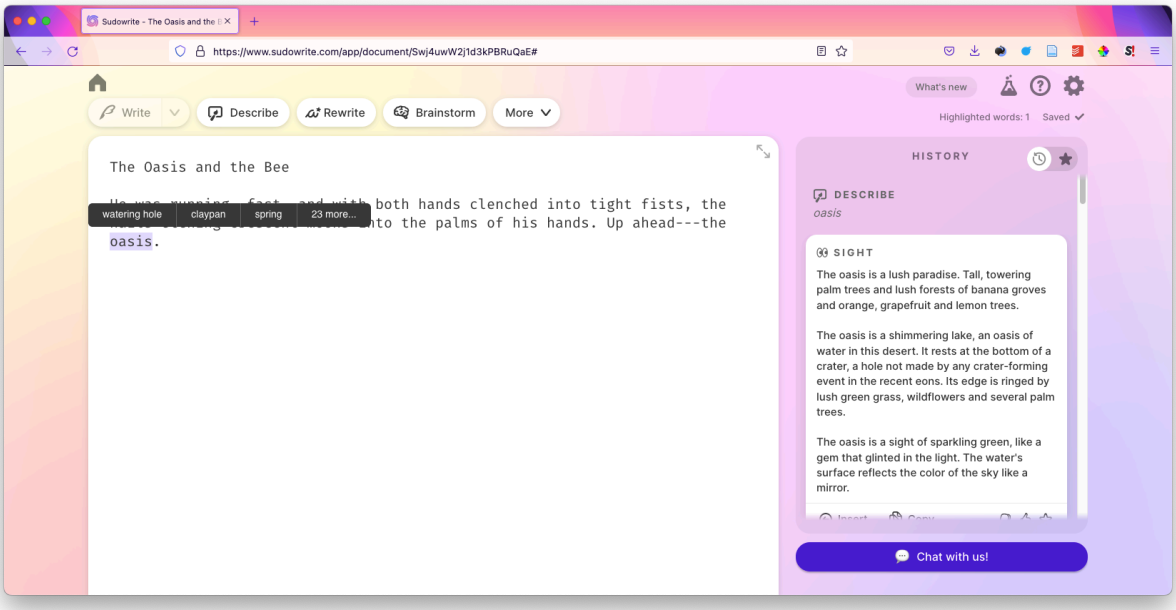


Figure C.2: 'Describe' feature of SudoWrite, which generates descriptions for highlight words or phrases.

tion, which seemed to be an earlier version of the 'write' function that is no longer available. These screenshots are intended to reflect SudoWrite functionality at the time the interviews were done.

C.1.2 Interview Questions

- **General questions about writing**

1. What kind of writing do you currently do? Have done in the past?
2. How long have you been writing?
3. What kind of formal or informal education have you had as a writer?
4. What is a piece of writing you are very proud of? Why?
5. Walk me through the process or life cycle for the last piece you wrote.

   Be sure to ask about external influences like research and feedback.
6. Do you ever get writer's block? What do you do?
7. Do you ever feel stuck revising something? What do you do?

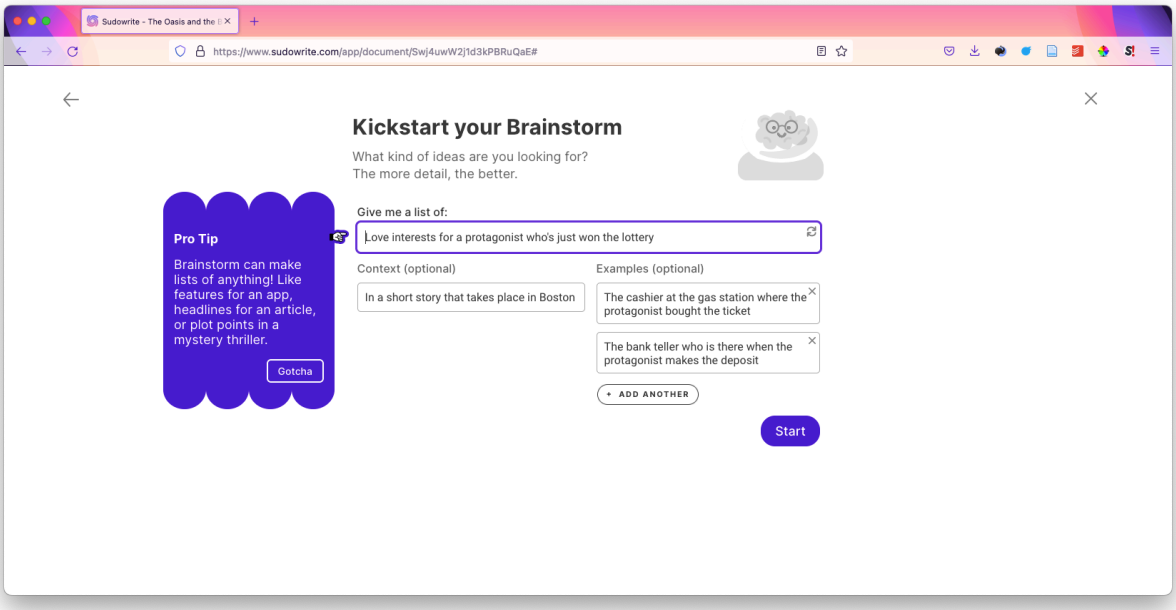


Figure C.3: 'Brainstorm' feature of SudoWrite, which generates ideas based on some context.

■ **Questions about existing influence**

8. Are there people who currently influence or in the past have influenced your writing? Who? In what ways?

   Does this happen abstractly (this teacher had a big impact on me) or more concretely (this teacher influences me when they give me feedback).

9. Are there texts or writers — who you haven't interacted with personally — who currently influence or in the past influenced your writing? Who/what? In what ways?

   Again, does this happen abstractly or concretely?

10. Someone giving you an assignment?
11. How about editors?
12. Do you use dictionaries, thesauruses, or other kinds of references when you write? Which? In what ways?
13. Have you done collaborative writing projects? Please share details.
14. Do you like to try new writing forms or styles? Why or why not?
15. Do you like to seek out feedback? Why or why not?

■ **Questions about computer influence**

Question template: If a computer program could _____ like a teacher or peer could, would you use it? Why or why not?

16. suggest places to revise
17. rewrite sections
18. write in a gap
19. finish a piece
20. continue something when you felt stuck
21. something about reader perspective – where a reader might get stuck or bored or confused
22. write into something out of domain – explanation or science fiction details
23. do research for you or describe a city or summarize a new technology

■ **Sudowrite user questions**

(Replaces 'Questions about computer influence')

24. What attracted you to SudoWrite?
25. Could you talk about a specific moment you used SudoWrite and what it looked like?
26. Which features do you use the most, and why?
27. In what ways do you feel like SudoWrite is 'human-like' or not in its capabilities?
28. Are there parts of your writing process you would not use SudoWrite (or something similar) for?
29. How do you feel SudoWrite impacts your writing?

    For example, how would your writing be different if you didn't use SudoWrite?
30. Does SudoWrite perform functions that otherwise might be performed by a peer, mentor, or editor?