Issue Brief

Controlling Large Language Model Outputs: A Primer

Authors
Jessica Ji
Josh A. Goldstein
Andrew J. Lohn

CSET CENTER *for* SECURITY *and* EMERGING TECHNOLOGY

December 2023

Executive Summary

Concerns over risks from generative artificial intelligence (AI) systems have increased significantly over the past year, driven in large part by the advent of increasingly capable large language models (LLMs). Many of these potential risks stem from these models producing undesirable outputs, from hate speech to information that could be put to malicious use. However, the inherent complexity of LLMs makes controlling or steering their outputs a considerable technical challenge.

This issue brief presents three broad categories of potentially harmful outputs—**inaccurate information, biased or toxic outputs, and outputs resulting from malicious use**—that may motivate developers to control LLMs. It also explains four popular techniques that developers currently use to control LLM outputs, categorized along various stages of the LLM development life cycle: **1) editing pre-training data**, **2) supervised fine-tuning**, **3) reinforcement learning with human feedback and Constitutional AI**, and **4) prompt and output controls**.

None of these techniques are perfect, and they are frequently used in concert with one another and with nontechnical controls such as content policies. Furthermore, the availability of open models—which anyone can download and modify for their own purposes—means that these controls or safeguards are unevenly distributed across various LLMs and AI-enabled products. Ultimately, this is a complex and novel problem that presents challenges for both policymakers and AI developers. Today's techniques are more like sledgehammers than scalpels, and even the most cutting-edge controls cannot guarantee that an LLM will never produce an undesirable output.

Introduction

Large language models (LLMs) are powerful AI models that can generate all kinds of text outputs, from poetry and professional emails to recipes and computer code. They have diffused broadly in recent months and are projected to have important societal ramifications. For example, venture capitalists and large tech companies alike have poured funding into LLM development and application-layer products built on top of these tools, and researchers expect LLMs to be broadly integrated into society and the economy in the years ahead.[1]

Despite their popularity and promise, LLMs are also capable of generating outputs that are hurtful, untrue, and, at times, even dangerous. In response to their increasing popularity, many people have asked: How do AI developers control the text that a language model generates?

In this primer, we tackle this question directly and present a high-level overview of how AI developers attempt to prevent LLMs from outputting harmful or otherwise undesirable text.* In the first section, we begin with a brief motivation of why developers seek to control or influence model outputs. In the second section, we describe some relevant features of the language model development pipeline before diving deeper into four classes of techniques that developers often use in practice in the third section. The fourth section provides context on why the open vs. private model paradigm further complicates developers' efforts to control and safeguard the outputs of their models.

A common theme across the various methods to limit harmful content is that none are perfect. Whether a particular intervention is successful frequently depends on how dedicated the user is to generating malicious or subversive text, and may be a matter of degree (how frequently the model produces undesirable text) rather than absolutes (whether it will produce the undesirable text at all).

* These practices can also be thought of as "aligning" LLMs with their developers' policies or guardrails. AI alignment is a broader research area that generally focuses on ensuring AI systems behave in ways that correspond to human values or human intentions.

Why Control Large Language Model Outputs?

Although interacting with a language model may occasionally feel like interacting with a real person, these systems are not human. Instead, language models are essentially complex probability-calculating machines. They establish relations between language tokens—words, phrases, parts of words, or even punctuation marks and grammatical symbols—and calculate the probability for each of them to come next in response to a given prompt. The models repeatedly choose one of the most likely tokens until their outputs are complete. Importantly, this means that language models have no underlying understanding of factualness or truthfulness, nor are they retrieving information from any single source. They are more akin to "improv machines"[2]: they excel at replicating patterns but have no built-in way to verify whether or not their outputs are useful, correct, or harmful.[3]

While a full taxonomy of risks falls outside the scope of this piece, we highlight some of the risks that motivate interest in controlling LLM outputs.[4] Firstly, some risks result from ordinary users simply receiving **incorrect information**. Users have already shown a propensity to misunderstand the limitations of these systems and inappropriately cite LLMs, thinking they provide factual information[5] (an example of what AI researchers call "overreliance"[6]). The range of potential failures is vast. Users depending on the system for health information who are fed false advice could put themselves at risk. Users turning to models for information about politics who receive false information may lose faith in candidates without justification, undermining the democratic process. As people use language models more frequently, the risks associated with overreliance will likely grow.

Secondly, content does not need to be demonstrably false to cause harm. This leads to another set of concerns that can occur when language models produce text that is **biased** (e.g., regarding race, gender, religion, or other categories) or **toxic**. Research has tested and found evidence of biases related to political ideology, religion, gender, and more in specific models.[7] Another line of research has traced biases in language models to the training data and noted that content excluded from training data based on certain keywords can disproportionately remove text from and about members of various minority groups.[8] Toxic content from LLMs may be particularly problematic if shown to children or other vulnerable groups.

Finally, there are also worries about bad actors using language models intentionally for **"malicious use."**[9] One worst-case scenario that has received public attention is the risk of a bad actor using a language model to learn how to create a homegrown bomb

or a bioweapon, although future research is needed to understand the relative risk (e.g., assessing the risk compared to information already available on Google).[10] Other troubling scenarios center on different types of malicious behavior such as the use of language models to facilitate hacking, scamming, or generating disinformation articles.[11] In all these cases, a strategy to prevent a model from outputting harmful materials would likely center on having it refuse to return such content outright.

How Large Language Models are Developed

As described above, language models are essentially complex probability-calculating machines. In order to understand how AI developers attempt to control their outputs, it is useful to first understand the process by which they are created, and how every stage of this process influences the system that ultimately ends up interacting with human end-users.

"Language model" is a general category describing a class of AI models that generate natural language text outputs. *Large* language models, or LLMs, are particularly powerful language models that are trained on especially large quantities of text, much of it scraped from the open internet.[12] While there is no strictly defined boundary between a regular language model and an LLM, a model's "largeness" is generally a question of scale, both in terms of data and computational power. Training an LLM is a long and computationally expensive process, with some of today's most cutting-edge models requiring thousands of powerful computer chips and hundreds of millions of dollars to create.[13] The most capable models are currently privately developed and protected as corporate intellectual property, but increasingly capable alternatives have been released openly for public use or adaptation.

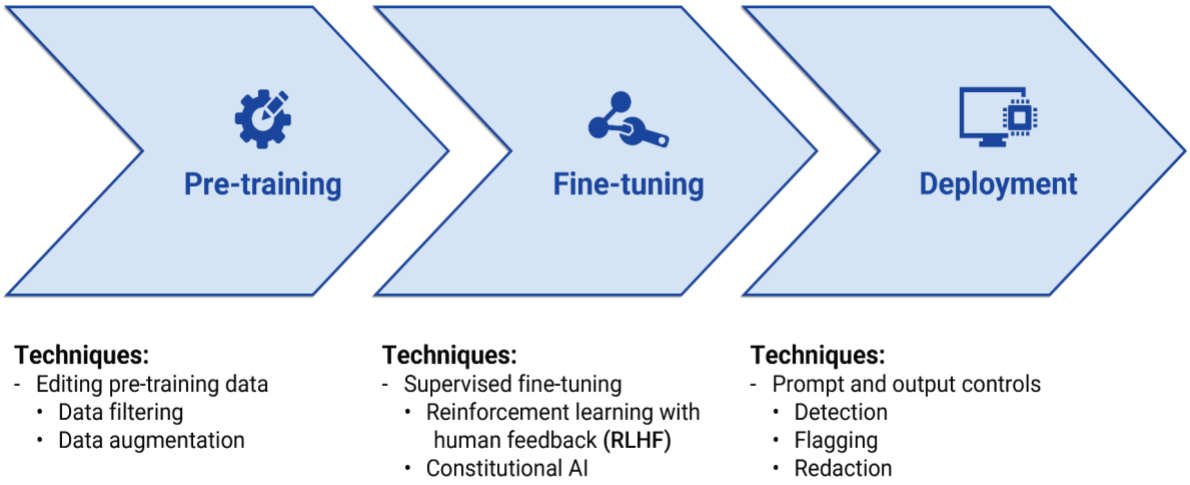
First, models are **pre-trained** on large general-purpose text datasets to learn correlations among tokens found in this natural language text. While some training datasets—consisting mostly of data scraped from publicly accessible web archives—are available for open inspection and use,[14] the exact composition of data sources used to train today's LLMs is largely unknown. Even AI developers generally do not have total visibility into the contents of their training datasets because the quantity of data required to pre-train an LLM is often in the scale of hundreds of terabytes.[15]

After this initial training, they are commonly **fine-tuned,** at least once, on smaller, more specialized datasets to improve their performance in a specific area. There are different types of fine-tuning for different purposes: reinforcement learning with human feedback attempts (described further in the following section) to guide models' behavior using input from humans, while other types of fine-tuning might train a model

more on data for a certain application or style in order to improve the model's capability to generate that kind of text specifically. These training steps are often repeated, with multiple rounds of iterative testing and evaluation to monitor model performance.

Finally, some fully-trained models are **deployed** for use, whether through a user-facing interface like a chatbot or via an API (application programming interface). The same model can be deployed in different forms; for example, OpenAI's GPT-4 has been deployed as both the LLM that powers ChatGPT and can also be accessed directly via its API, which allows third-party developers to integrate it into their software products without having direct access to the model. Another popular option for developers is to open-source their model, which allows anyone to access its underlying code, fine-tune it to their own specifications, and use it to build their own applications.

Figure 1: Stages in the AI development pipeline and some associated language model control techniques.



Source: CSET.

Each of these steps in the development process offers opportunities to shape the model's behavior. In the following section, we discuss four classes of techniques that LLM developers use to steer outputs.

Four Techniques to Control LLM Outputs

1. Editing Pre-training Data

Since language models' predictive power derives from correlations in the text they are trained on, a common misconception about LLMs is that their outputs can be easily steered by manipulating or editing their training data. A model whose training data contains no reference to a particular word, for instance, is extremely unlikely to output anything containing that word. However, real-world pre-training is much more complicated. Considering the sheer volume of data that these models are pre-trained on, it is extremely difficult to predict how changing their training data will affect their performance or their propensity to output certain types of content.

For example, one study found that filtering a dataset reduced one language model's likelihood of generating harmful text, but the filtered versions also consistently performed worse on standard performance benchmarks than their unfiltered counterparts.[16] Many LLM developers are wary of techniques that decrease model performance, especially if other techniques can also be used to control outputs. Data filtering methods can also backfire and lead to other unwanted outcomes, like erasing dialect patterns or marginalizing certain groups within the model's training data.[17] And while data augmentation—supplementing training data with examples of desired outcomes—has shown some promise, it is very difficult to effectively scale in order to reduce bias in large models.[18]

Ultimately, while training data manipulation is theoretically a powerful mechanism to control model behavior, it is not a panacea for preventing many types of harmful output, especially when meaning and harm are context dependent.[19] Even though factors like content filters and data sources can ultimately have significant effects on the fully trained model's behavior,[20] researchers have yet to fully understand exactly how to manipulate data in a way that will have meaningful impacts on the resulting model while minimizing performance loss. Smaller, specialized language models that are pre-trained on curated datasets are likely to have more success with data filtration or augmentation, but LLM developers are likely to rely on other methods in order to steer their models.

2. Supervised Fine-Tuning

Once a model has been pre-trained, developers can continue to adjust its behavior by training it further on a specialized dataset. This process, known as supervised fine-tuning, is one of the most common ways to modify a language model, usually in an effort to improve its performance in a particular area. To make a general-purpose model like OpenAI's GPT-4 better at math, for instance, an intuitive solution is to train the model on math problems to improve its ability to recognize patterns in that particular domain. The more high-quality data a model has been exposed to that is relevant to a specific topic, the better the model will be at predicting the next token in its output in a way that is likely to be useful to human users.

Supervised fine-tuning can be quite powerful in the right context when the right kind of data is available, and is one of the best ways to specialize a model to a specific domain or use case. ("Supervised," in this context, refers to the fact that the model is provided with labeled data and thus does not have to perform the prerequisite step of learning patterns and associations within the data.) For example, fine-tuning an LLM on a collection of datasets described via instructions—consisting of specific tasks or requests, such as "can we infer the following?" or "translate this text into Spanish"—significantly improves the model's ability to respond to prompts in natural language.[21] This technique, now known as instruction tuning, has been instrumental in creating LLM chatbots that can interpret all different kinds of inputs, from simple questions and declarative statements to lists of multi-step instructions.

Fine-tuning is a broadly applicable specialization technique that can be applied in certain contexts to steer model behavior. Some research has shown that this technique can not only improve a model's performance in a particular area[22] but can also compensate for bias inherited from the pretrained model.[23] These biases, which can include those related to protected categories like race or gender, stem from statistical patterns that the model has learned from its training data. A training dataset that only consists of images of male doctors, for example, will result in a model that will consistently produce images of men when prompted with "doctor." Fine-tuning the model on a more balanced dataset can be one way to correct this issue.

However, effective supervised fine-tuning depends on access to specialized and high-quality datasets, which may not be available in all domains or accurately capture the behavior that researchers are attempting to control. Researchers have therefore looked to develop alternative techniques that are either not as reliant on specialized data or that allow for a more flexible way to steer an LLM's behavior.

3. Reinforcement Learning with Human Feedback (RLHF) and Constitutional AI

Two techniques often used to complement supervised fine-tuning employ reinforcement learning, which is the process of training a machine learning model to make decisions via many iterations of trial and error. Over the course of the training process, the model receives either negative or positive feedback which gradually "teaches" it to take the series of actions that will maximize the amount of positive feedback. Given the right conditions, reinforcement learning can be extremely effective and powerful—Google DeepMind's AlphaGo Zero system, an earlier version of which famously beat the human Go world champion in 2016,[24] was primarily trained using reinforcement learning. While playing millions of games against itself over the course of several days, AlphaGo Zero repeatedly updated its own parameters to select better and better moves.[25]

Since reinforcement learning already incorporates a built-in feedback process, AI researchers leveraged it to create new techniques for fine-tuning LLMs. Reinforcement learning with human feedback (RLHF) is a technique in which an LLM is fine-tuned with the help of a different machine-learning model, known as a "reward model." This reward model is trained on some of the original LLM's text outputs, which human annotators have ranked based on some set of guidelines or preferences. The goal of the reward model is to encode human preferences: once given some text input, it outputs a numerical score which reflects how likely humans are to prefer that text. The reward model then serves as the basis for the feedback mechanism used to fine-tune the original LLM: as the original LLM outputs text, the reward model "scores" that output and with each iteration, the original LLM adjusts its output to improve its "score."[26] Put more simply, RLHF attempts to train an LLM to generate outputs that humans are more likely to deem acceptable. It's perhaps most well-known for its role in turning OpenAI's GPT-3.5 into ChatGPT,[27] and has been remarkably successful at producing LLMs that interact with users in human-like ways.

Unlike supervised fine-tuning, which is typically used for creating a specialized model and does not necessarily involve steering a model based on any sense of "right" or "wrong," RLHF centers on the principle that human preferences should play a role in how an LLM behaves. The "human feedback" aspect of RLHF is its central component and also its greatest limitation. For example, in 2022 a team of OpenAI researchers hired 40 contractors to create and label a dataset of human preferences.[28] Today, data annotators around the world spend hours rating interactions with pre-deployed versions of AI systems like ChatGPT and Google's Sparrow.[29] As long as human labor is necessary for RLHF, LLM creators will naturally face limitations on how much human

feedback their models will receive because of the sheer time and cost of such measures.[30] Furthermore, RLHF is tricky to implement even with enough feedback. A poorly designed feedback process *may* result in the model learning how to act in ways that maximize the amount of positive feedback it receives but that may not actually translate into the kinds of outputs that human users prefer.[31]

Constitutional AI is a related fine-tuning process, developed by the AI company Anthropic, that attempts to steer an LLM's behavior with minimal human guidance.[32] Unlike RLHF, Constitutional AI does not rely on human labels or annotations as a way to encode human preferences. Instead, researchers provide the system with a list of guiding rules or principles—hence the term "constitutional"—and essentially ask another model to evaluate and revise its outputs. [33] While Constitutional AI is promising as an RLHF alternative that relies on far fewer human-generated labels, RLHF still seems to be the industry standard for guiding and steering LLMs at the fine-tuning stage.[34]

4. Prompt and Output Controls

Even after pre-training and multiple rounds of fine-tuning, an LLM may still output undesirable text. Before developers incorporate models into consumer-facing products, they can choose to control models using additional techniques at either the pre-output or the post-output stage. These techniques are also commonly referred to as "input filters" (applied at the pre-output stage) and "output filters" (applied at the post-output stage) and generally fall into three camps: detection, flagging, and redaction. Many existing tools and solutions originate from the need to moderate content on social media and are not necessarily specific to AI, but developers have increasingly adapted them for use on large language models.[35]

Before the LLM even receives a user's input, developers can screen prompts to assess whether they are likely to evoke harmful text and show users a warning or refusal message in lieu of completion from the AI system. This can create a similar effect to the model itself refusing to answer certain types of prompts. While both of these methods can be bypassed by jailbreaking, in which users deliberately circumvent models' content restrictions,[36] these methods can serve as a basic defense for non-malicious users. AI developers are also increasingly evaluating their LLMs using "red-teaming," a systematic testing process that includes jailbreaking models in a controlled environment in order to see how their guardrails might fail.[37] While red-teaming is used for a number of purposes, including acquiring information that can be used to improve controls, one major incentive for developers is to detect and fix potential

jailbreaks before the model is released to the public. Another variation on prompt screening is prompt rewriting, in which a different language model rewrites user-submitted prompts before they are passed to the target model.[38] This may provide some protection against jailbreaking, as it effectively creates another set of guardrails between the user and the target model.

At the post-output stage, once the LLM has composed a response to a prompt but before that output has been shown to the user, developers can employ additional checks and filters. One option is to train a separate machine learning model—often referred to as a "toxicity filter"[39]—to detect harmful content, then use that model to catch outputs before they can be shown to users. Like supervised fine-tuning, these techniques rely on human-labeled data. While they have demonstrably positive effects on how toxic LLM outputs are, labeling the datasets of harmful content that are used to train the detection models is often actively harmful to workers' mental health.[40]

Post-fine-tuning model controls are also often combined with monitoring or user reporting. Usually, this involves a combination of automated content detection or filtering, human content moderation, and user reporting. For example, OpenAI provides a moderation endpoint that developers can use to automatically flag or filter potentially harmful outputs,[41] and was also initially relying on human content moderators to help evaluate outputs from ChatGPT.[42] Finally, if a harmful or undesirable output makes it through all of the existing controls, many LLM interfaces contain a user feedback mechanism so that users can flag individual outputs directly. Developers are extremely unlikely to catch every prompt or use case that might lead to a harmful output, and thus rely on users to give feedback on model performance.

Open vs. Private Models

The safeguards listed in the previous section are mostly voluntary techniques that are applied by the companies that develop and host top-tier LLMs. These large companies generally have both the incentives and the resources to attempt to secure their models and improve the safety of their outputs. For example, minimizing the chances that a language model will return toxic text may improve the user experience and boost public confidence in the model, making it easier to incorporate into a software product.

However, private companies are not the only ones building and supplying these models. Smaller, but still powerful, open models are increasingly available for anyone to download and adapt. Some publicly accessible models were originally trained by AI labs on carefully selected pre-training data and may have been red-teamed prior to release, but these released models can be fine-tuned by third-party users who may be

less diligent or even outright malicious. Other open models may be trained by developers who may have fewer resources dedicated to safety, less interest in curating their fine-tuning data, or fewer incentives to monitor the prompts that are provided to their models or the outputs that are ultimately created. Recently, some powerful open models have also been produced in foreign countries, such as the United Arab Emirates and China, where developers may have different views about what guardrails should exist.[43] While many of these open models are created with safety and responsible use in mind, there is no way to guarantee that all of their downstream users adhere to the same standards and that the resulting applications will be error-free or sufficiently safeguarded. In fact, some research suggests that fine-tuning a model can undermine its developers' safeguards even when users may not intend to do so.[44]

The AI development community is currently debating whether private or open models are better for safety. For one, private models are not guaranteed to be easier to control in all circumstances. Even if they are secured and safeguarded, cutting-edge models are more likely to possess capabilities that require novel or more rigorous control techniques. Other variables, such as whether or not the user is interfacing directly with the model, may also affect how easy it is to control. Finally, while open models are difficult to control and monitor once they are adopted by downstream users, they also broaden access to researchers outside private companies who may have fewer resources or need the flexibility to experiment freely with an LLM.

Conclusion

Controlling LLM outputs remains challenging. In practice, the methods above are almost always used in combination with each other, and undesirable outputs will continue to slip through despite developers' best efforts. Today's methods are more like sledgehammers than scalpels; any attempt to control a model in a specific way, such as preventing it from outputting violent content, is likely to have unintended consequences, like making it unable to describe the plot of an R-rated movie.[45] There are also legitimate disagreements about whether or not a given output is harmful, and the definition of what constitutes harmful or toxic content might vary depending on the context in which any given model is deployed.

Several other factors complicate the situation further. Firstly, this is a pacing problem. AI researchers are racing to develop and test these techniques while simultaneously keeping up with the breakneck pace of AI capabilities progress. The popularity of jailbreaks and other methods for bypassing content controls also means that

developers are constantly discovering new ways their models can be manipulated. Finally, it is very difficult for those outside the leading AI labs to evaluate how effective these individual methods are because there is little information about their effectiveness for some of the most popular and powerful LLMs. While open models can provide useful data in this vein, they may be smaller and less capable than state-of-the-art models. Public data on user behavior, such as API calls or what kinds of feedback users are giving models, is also scarce.

Language models can carry inherent risks, including their propensity to output undesirable text, including falsehoods, potentially dangerous information such as instructions for biological or nuclear weapons, or malware code. Nevertheless, the idea that developers can gain perfect control over an LLM by simply tweaking its inputs is misleading. LLMs can be complex, messy, and behave in unpredictable ways. As AI governance and regulation become increasingly important, however, understanding how they work and how they might be controlled will be more critical than ever.

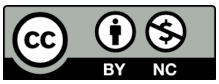
Authors

Jessica Ji is a Research Analyst with the CyberAI Project at Georgetown University's Center for Security and Emerging Technology (CSET), where Josh Goldstein is a Research Fellow.

Andrew Lohn is a Senior Fellow at CSET and the Director for Emerging Technology on the National Security Council Staff, Executive Office of the President under an Interdepartmental Personnel Act agreement with CSET. Dr. Lohn completed this work before starting at the National Security Council. The views expressed are the author's own personal views and do not necessarily reflect the views of the White House or the Administration. During the preparation of this brief, Andrew Lohn also participated in the red-teaming of Meta's and OpenAI's large language models for which he was compensated.

Acknowledgments

For feedback and assistance, we would like to thank John Bansemer, Helen Toner, Karson Elmgren, Krystal Jackson, Rishi Bommasani, and Katherine Lee. We are also grateful to Lauren Lassiter, Shelton Fitch, Tessa Baker, and Jason Ly for editorial assistance.



Document Identifier: doi: 10.51593/2023CA009

Endnotes

[1] Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock, "GPTs are GPTs: An Early Look at the Labor Market Impact Potential of Large Language Models," arXiv preprint arXiv:2303.10130v5 (2023), https://arxiv.org/abs/2303.10130.

[2] Helen Toner, "AI Chatbots Are Doing Something a Lot Like Improv," *TIME*, May 18, 2023, https://time.com/6280533/ai-chatbots-improv-machines/.

[3] Some researchers are exploring different ways to ground language model outputs in factual sources. For more, see Jacob Menick, Maja Trebacz, Vladimir Mikulik, John Aslanides, Francis Song, Martin Chadwick et al., "Teaching language models to support answers with verified quotes," arXiv preprint arXiv:2203.11147 (2022), https://arxiv.org/abs/2203.11147.

[4] For a broader taxonomy of risks from LLMs, see: Laura Weidinger, Jonathan Uesato, Maribeth Rauh, Conor Griffin, Po-Sen Huang, John Mellor, Amelia Glaese, Myra Cheng, et al., "Taxonomy of Risks posed by Language Models," FAccT '22: Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency, June 2022, https://dl.acm.org/doi/10.1145/3531146.3533088.

[5] Benjamin Weiser, "Here's What Happens When Your Lawyer Uses ChatGPT," *The New York Times*, May 27, 2023, https://www.nytimes.com/2023/05/27/nyregion/avianca-airline-lawsuit-chatgpt.html.

[6] OpenAI, "GPT-4 Technical Report," arXiv preprint arXiv:2303.08774 (2023), https://arxiv.org/abs/2303.08774.

[7] Jochen Hartmann, Jasper Schwenzow, and Maximilian Witte, "The political ideology of conversational AI: Converging evidence on ChatGPT's pro-environmental, left-libertarian orientation," arXiv preprint arXiv:2301.01768 (2023), https://arxiv.org/abs/2301.01768; Abubakar Abid, Maheen Farooqi, and James Zou, "Persistent Anti-Muslim Bias in Large Language Models," AIES '21: Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society, July 30, 2021, https://dl.acm.org/doi/pdf/10.1145/3461702.3462624; Umang Gupta, Jwala Dhamala, Varun Kumar, Apurv Verma, Yada Pruksachatkun, Satyapriya Krishna, Rahul Gupta, Kai-Wei Chang, Greg Ver Steeg, and Aram Galstyan, "Mitigating Gender Bias in Distilled Language Models via Counterfactual Role Reversal," arXiv preprint arXiv:2203.12574 (2022), https://arxiv.org/abs/2203.12574; for a more thorough discussion of language models and bias, see Emilio Ferrara, "Should ChatGPT be Biased? Challenges and Risks of Bias in Large Language Models," arXiv preprint arXiv:2304.03738 (2023), https://arxiv.org/abs/2304.03738.

[8] Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner, "Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus," Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processes, November 7-11, 2021, https://aclanthology.org/2021.emnlp-main.98.pdf.

[9] Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang et al., "Ethical and social risks of harm from Language Models," arXiv preprint arXiv:2112.04359 (2021), https://arxiv.org/abs/2112.04359; Miles Brundage, Shahar Avin, Jack Clark, Helen Toner, Peter Eckersley, Ben Garfinkel et al., "The Malicious Use of Artificial Intelligence: Forecasting, Prevention, and Mitigation," arXiv preprint arXiv:1802.07228 (2018), https://arxiv.org/abs/1802.07228.

[10] OpenAI, "GPT-4 Technical Report."

[11] Julian Hazell, "Large Language Models Can Be Used To Effectively Scale Spear Phishing Campaigns," arXiv preprint arXiv:2305.06972 (2023), https://arxiv.org/abs/2305.06972; Andrew J. Lohn and Krystal A. Jackson, "Will AI Make Cyber Swords or Shields?," Center for Security and Emerging Technology, August 2022, https://cset.georgetown.edu/publication/will-ai-make-cyber-swords-or-shields/; Josh A. Goldstein, Jason Chao, Shelby Grossman, Alex Stamos, and Michael Tomz, "Can AI Write Persuasive Propaganda?," SocArXiv preprint, April 8, 2023, https://doi.org/10.31235/osf.io/fp87b.

[12] Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner, "Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus," Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, November 2021, https://aclanthology.org/2021.emnlp-main.98/.

[13] Will Knight, "OpenAI's CEO Says the Age of Giant AI Models Is Already Over," Wired, April 17, 2023, https://www.wired.com/story/openai-ceo-sam-altman-the-age-of-giant-ai-models-is-already-over/.

[14] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He et al., "The Pile: An 800GB Dataset of Diverse Text for Language Modeling," arXiv preprint arXiv:2101.00027 (2020), https://arxiv.org/abs/2101.00027.

[15] Kevin Schaul, Szu Yu Chen and Nitasha Tiku, "Inside the secret list of websites that make AI like ChatGPT sound smart," The Washington Post, April 19, 2023, https://www.washingtonpost.com/technology/interactive/2023/ai-chatbot-learning/.

[16] Helen Ngo, Cooper Raterink, João G.M. Araújo, Ivan Zhang, Carol Chen, Adrien Morisot, and Nicholas Frost, "Mitigating harm in language models with conditional-likelihood filtration," arXiv preprint arXiv:2108.07790 (2021), https://arxiv.org/abs/2108.07790.

[17] Sachin Kumar, Vidhisha Balachandran, Lucille Njoo, Antonios Anastasopoulos, and Yulia Tsvetkov, "Language Generation Models Can Cause Harm: So What Can We Do About It? An Actionable Survey," arXiv preprint arXiv:2210.07700 (2023), https://arxiv.org/abs/2210.07700.

[18] Emily Dinan, Angela Fan, Adina Williams, Jack Urbanek, Douwe Kiela, and Jason Weston, "Queens are Powerful too: Mitigating Gender Bias in Dialogue Generation," arXiv preprint arXiv:1911.03842 (2020), https://arxiv.org/abs/1911.03842.

[19] Betty van Aken, Julian Risch, Ralf Krestel, Alexander Löser, "Challenges for Toxic Comment Classification: An In-Depth Error Analysis," arXiv preprint arXiv:1809.07572 (2018), https://arxiv.org/abs/1809.07572.

[20] Shayne Longpre, Gregory Yauney, Emily Reif, Katherine Lee, Adam Roberts, Barret Zoph, Denny Zhou, and Jason Wei et al., "A Pretrainer's Guide to Training Data: Measuring the Effects of Data Age, Domain Coverage, Quality, & Toxicity," arXiv preprint arXiv:2305.13169 (2023), https://arxiv.org/abs/2305.13169.

[21] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, Quoc V. Le, "Finetuned Language Models Are Zero-Shot Learners," arXiv preprint arXiv:2109.01652 (2022), https://arxiv.org/abs/2109.01652.

[22] Tiedong Liu and Bryan Kian Hsiang Low, "Goat: Fine-tuned LLaMA Outperforms GPT-4 on Arithmetic Tasks," arXiv preprint arXiv:2305.14201 (2023), https://arxiv.org/abs/2305.14201.

[23] Angelina Wang and Olga Russakovsky, "Overcoming Bias in Pretrained Models by Manipulating the Finetuning Dataset," arXiv preprint arXiv:2303.06167 (2023), https://arxiv.org/abs/2303.06167.

[24] "AlphaGo," Google DeepMind, accessed July 2023, https://www.deepmind.com/research/highlighted-research/alphago.

[25] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang et al., "Mastering the game of Go without human knowledge," *Nature* 550, 354–359 (2017), https://doi.org/10.1038/nature24270.

[26] Nathan Lambert, Louis Castricato, Leandro von Werra, and Alex Havrilla, "Illustrating Reinforcement Learning from Human Feedback (RLHF)," *Hugging Face Blog*, December 9, 2022, https://huggingface.co/blog/rlhf.

[27] "Introducing ChatGPT," *OpenAI Blog*, November 30, 2022, https://openai.com/blog/chatgpt.

[28] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang et al., "Training language models to follow instructions with human feedback," arXiv preprint arXiv:2203.02155 (2022), https://arxiv.org/abs/2203.02155.

[29] Josh Dzieza, "AI Is a Lot of Work," *The Verge*, June 20, 2023, https://www.theverge.com/features/23764584/ai-artificial-intelligence-data-notation-labor-scale-surge-remotasks-openai-chatbots.

[30] Some researchers are exploring alternatives to RLHF that involve simulating human feedback using LLMs. See Yann Dubois, Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang et al., "AlpacaFarm: A Simulation Framework for Methods that Learn from Human Feedback," arXiv preprint arXiv:2305.14387 (2023), https://arxiv.org/abs/2305.14387.

[31] Rohin Shah, Vikrant Varma, Ramana Kumar, Mary Phuong, Victoria Krakovna, Jonathan Uesato, Zac Kenton, "Goal Misgeneralization: Why Correct Specifications Aren't Enough For Correct Goals," arXiv preprint arXiv:2210.01790 (2022), https://arxiv.org/abs/2210.01790.

[32] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen et al., "Constitutional AI: Harmlessness from AI Feedback," arXiv preprint arXiv:2212.08073 (2022), https://arxiv.org/abs/2212.08073.

[33] "Claude's Constitution," Anthropic, May 9, 2023, https://www.anthropic.com/index/claudes-constitution.

[34] Widespread industry adoption of RLHF has spurred further research into how human feedback might be incorporated into other stages of the development pipeline, such as the pre-training stage. For more, see Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Bhalerao, Christopher L. Buckley, Jason Phang, Samuel R. Bowman, Ethan Perez, "Pretraining Language Models with Human Preferences," arXiv preprint arXiv:2302.08582 (2023), https://arxiv.org/abs/2302.08582.

[35] Alex Pasternack, "Google's Jigsaw was trying to fight toxic speech with AI. Then the AI started talking," *Fast Company*, July 31, 2023, https://www.fastcompany.com/90929549/google-jigsaw-toxic-speech-ai.

[36] Yi Liu, Gelei Deng, Zhengzi Xu, Yuekang Li, Yaowen Zheng, Ying Zhang, Lida Zhao, Tianwei Zhang, and Yang Liu, "Jailbreaking ChatGPT via Prompt Engineering: An Empirical Study," arXiv preprint arXiv:2305.13860 (2023), https://arxiv.org/abs/2305.13860.

[37] Nazneen Rajani, Nathan Lambert, and Lewis Tunstall, "Red-Teaming Large Language Models," *Hugging Face Blog*, February 24, 2023, https://huggingface.co/blog/red-teaming.

[38] OpenAI, "DALL-E 3 System Card," OpenAI Research, October 3, 2023, https://openai.com/research/dall-e-3-system-card.

[39] Johannes Welbl, Amelia Glaese, Jonathan Uesato, Sumanth Dathathri, John Mellor, Lisa Anne Hendricks, Kirsty Anderson, Pushmeet Kohli, Ben Coppin, and Po-Sen Huang, "Challenges in

Detoxifying Language Models," arXiv preprint arXiv:2109.07445 (2021), https://arxiv.org/abs/2109.07445.

[40] Billy Perrigo, "Exclusive: OpenAI Used Kenyan Workers on Less Than $2 Per Hour to Make ChatGPT Less Toxic," *Time Magazine*, January 18, 2023, https://time.com/6247678/openai-chatgpt-kenya-workers/.

[41] OpenAI Platform, "Moderation," accessed August 1, 2023, https://platform.openai.com/docs/guides/moderation/.

[42] Reed Albergotti, "Breakthrough AI tools still rely on old-school moderation techniques," *Semafor*, December 14, 2022, https://www.semafor.com/article/12/14/2022/breakthrough-ai-tools-still-rely-on-old-school-moderation-techniques.

[43] Technology Innovation Institute, "Falcon-180B," Hugging Face (2023), https://huggingface.co/tiiuae/falcon-180B; Aiyuan Yang et. al., "Baichuan 2: Open Large-scale Language Models," arXiv preprint arXiv:2309.10305 (2023), https://arxiv.org/abs/2309.10305.

[44] Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson, "Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To!," arXiv preprint arXiv:2310.03693 (2023), https://arxiv.org/abs/2310.03693.

[45] Andreas Kling (@awesomekling), 2023, "The new Bing Chat AI keeps regretting answering my questions, and removes its own messages after reconsidering." X, February 26, 2023, 12:30 p.m. https://twitter.com/awesomekling/status/1629896620492898305?lang=en.