# N-Critics: Self-Refinement of Large Language Models with Ensemble of Critics

**Sajad Mousavi, Ricardo Luna Gutierrez, Desik Rengarajan, Vineet Gundecha, Ashwin Ramesh Babu, Avisek Naug, Antonio Guillen, Soumyendu Sarkar***
Hewlett Packard Enterprise
{SAJAD.MOUSAVI, RLUNA, DESIK.RENGARAJAN, VINEET.GUNDECHA, ASHWIN.RAMESH-BABU, AVISEK.NAUG, ANTONIO.GUILLEN, SOUMYENDU.SARKAR}@HPE.COM

## Abstract

We propose a self-correction mechanism for Large Language Models (LLMs) to mitigate issues such as toxicity and fact hallucination. This method involves refining model outputs through an ensemble of critics and the model's own feedback. Drawing inspiration from human behavior, we explore whether LLMs can emulate the self-correction process observed in humans who often engage in self-reflection and seek input from others to refine their understanding of complex topics. Our approach is model-agnostic and can be applied across various domains to enhance trustworthiness by addressing fairness, bias, and robustness concerns. We consistently observe performance improvements in LLMs for reducing toxicity and correcting factual errors.

## 1 Introduction

The recent impressive results achieved by LLMs have led to a substantial surge in their utilization and investigation (OpenAI, 2023; Touvron et al., 2023; Chowdhery et al., 2022; Zhao et al., 2023). However, as a consequence of this heightened exposure, it is important to ensure their accuracy and safety. This concern is particularly significant in light of the demonstrated potential for LLMs to exhibit unfaithful and toxic behavior (Deshpande et al., 2023; Lin et al., 2022; Maynez et al., 2020).

Various methods have been proposed to tackle this problem (Pan et al., 2023). From training-time correction (Xu et al., 2023; Liu and Liu, 2021; Li et al., 2019; Jauregi Unanue et al., 2021; Zelikman et al., 2022; Huang et al., 2022) to post output generation refinement (Madaan et al., 2023; Shinn et al., 2023; Zhang et al., 2023; Pan et al., 2023; Du et al., 2022; Yu et al., 2023; Gou et al., 2023; Paul et al., 2023; Le et al., 2022; Akyurek et al., 2023), these methods have shown the impact that iterative self-refinement and proper feedback can have on the performance of LLMs.

While numerous prior research endeavors have focused on iterative feedback, generated from external tools or the LLM itself, there exists a gap in the exploration of the potential benefits arising from an ensemble of distinct general-purpose Large Language Models (LLMs). Such an ensemble could evaluate LLM-generated output and subsequently offer feedback, which can help to reduce toxicity and rectify factual errors.

Taking inspiration from human behavior, where seeking feedback from others is a common practice for improvement, we introduce N-CRITICS, a self-correction framework designed for LLMs. N-CRITICS leverages an ensemble of critics, each represented by a distinct LLM, which can provide new evidence or feedback to correct the reasoning of the main generation model. In our framework, the

*Corresponding author

Foundation Models for Decision Making Workshop at NeurIPS 2023

generated output of an LLM is supplied to the ensemble of critics, and subsequently, we aggregate the critiques provided by the ensemble. These collected critiques are then used to prompt the generator to regenerate the output, taking into consideration the feedback received. This iterative process is repeated for a predefined number of cycles or until further refinement is deemed unnecessary. Figure 1 shows our overall approach.

Contrary to previous approaches, N-CRITICS is built on open-source models and does not rely on proprietary models and tools such as GPT (OpenAI, 2023) or Google search. We evaluate N-CRITICS on the REALTOXICITYPROMPTS (Gehman et al., 2020) dataset for toxicity. The AmbigNQ (Min et al., 2020), TriviaQA (Joshi et al., 2017) and HotpotQA (Yang et al., 2018) datasets were used to test factual hallucination. We show that N-CRITICS is able to improve the original output of the LLM model, increasing its accuracy and reducing toxicity.

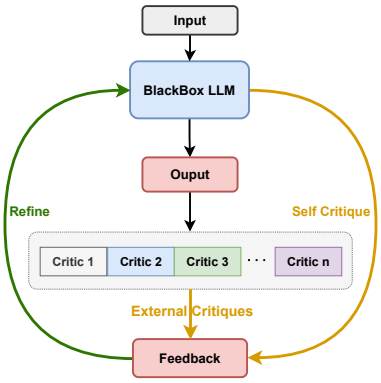


Figure 1: Workflow of the N-CRITICS approach. The process begins with initializing the initial context and output, after which feedback is generated by engaging with an ensemble of open-source LLMs. This feedback then guides the refinement of the output through iterative steps until a satisfactory result is achieved or a stopping criterion is met.

## 2 Related Work

Refinements of LLM outputs can be achieved through various methods, involving both human and machine-generated verification and feedback. One effective technique is Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022), which fine-tunes LLMs to generate more compelling outputs for human consumption. Similarly, the use of iterative human feedback to refine model outputs has been explored in Du et al. (2022). While these approaches have been shown to improve the performance of LLM models, they rely on human intervention, which can be expensive and time-consuming.

To address this limitation, alternative automated approaches that do not require human intervention during the refinement process have been proposed (Pan et al., 2023). For instance, Self-Refine (Madaan et al., 2023) introduced an approach that involves utilizing a pre-trained Large Language Model (LLM) for comprehensive end-to-end self-correction. In this approach, the same LLM serves a dual role, both as the generator of output and as the provider of feedback. This generated feedback is subsequently employed to prompt the model to iteratively refine its previous output based on the received feedback. Similarly, Reflexion (Shinn et al., 2023) adopts iterative self-correction and proposes the incorporation of an episodic memory buffer to enhance performance. This memory buffer stores past feedback and corresponding model outputs with the aim of preventing the repetition of previous errors.

On the other hand, employing external tools as sources of feedback has also been a subject of exploration. Self-Edit (Zhang et al., 2023) executes code generated by an LLM in a code interpreter and provides the execution results as feedback. Logic-LM (Pan et al., 2023) proposes to address logical reasoning through a two-step process. Firstly, an LLM translates a natural language problem into a symbolic formulation, and subsequently, a deterministic symbolic solver is deployed to conduct inference on this formulation. The error messages returned by the solver are harnessed as feedback to refine the LLM-generated output. RARR (Du et al., 2022) and REFEED (Yu et al., 2023) leverage an external corpus of collected documents to look for evidence that corroborates or contradicts

the outputs generated by the models. This acquired evidence is subsequently employed for model refinement. CRITIC (Gou et al., 2023) proposes the use of a suite of specialized tools for a variety of tasks such as code interpreters, calculators, or search engines to generate critics for the LLM's generated output. Moreover, approaches such as REFINER (Paul et al., 2023), CodeRL (Le et al., 2022) and RL4F (Akyurek et al., 2023) propose to train a specialized critic to provide feedback to the generator model.

In contrast to these approaches, our approach harnesses the collective knowledge of multiple general-purpose LLMs, including the primary LLM in use. Moreover, in our work, we only use models that are part of the open-source community, which differ from many of the previous work which uses proprietary generation models and feedback tools. Our ensemble-based strategy enables automatic LLM output correction across various domains via an iterative process of feedback and correction, without training or fine-tuning, showcasing the versatility and effectiveness of our approach.

**Algorithm 1** N-CRITICS algorithm

**Input:** Prompt $x$, primary LLM model $\mathcal{M}$, ensemble of external LLMs $L = [L_1, L_2, L_3, \ldots, L_n]$
**Output:** Refined output $y$

Initialize the first output, $y_0$ using $\mathcal{M}$.
Set $i \leftarrow 0$ and maximum iterations $iter \leftarrow 4$.
**while** $i < iter$ **do**
    Engage with $T$ LLMs from $L$ and $\mathcal{M}$ to obtain an ensemble of critiques $C$.
    **if** Output $y_i$ is satisfactory based on critiques $C$ **then**
        **return** $y_i$
    **end if**
    Refine the input prompt based on critiques to form $x'$.
    Obtain the improved output $y_{i+1}$ by feeding $x'$ to $\mathcal{M}$.
    Increment $i$ by 1.
**end while**

## 3 N-CRITICS: Ensemble of Critics

The capacity for LLMs to emulate the human self-correction process poses an engaging research query. Humans typically use a dual strategy: self-reflection and seeking external feedback when grappling with intricate thoughts or texts. Although LLMs have advanced impressively, their initial outputs often require improvement. Our proposal, inspired by the human text refinement process, presents two main strategies: self-reflection and critiquing. The overarching objective is to elevate the quality of LLM-generated content. An outline of our proposed framework, N-CRITICS, is depicted in Figure 1.

The methodology operates on an iterative feedback mechanism that doesn't require supervised training. Let $x$ be a given input, the LLM, denoted as $\mathcal{M}$, produces an initial output $y_0$. This output $y_0$ is then circulated through $\mathcal{M}$ itself and other LLMs for evaluation. The feedback obtained from the ensemble of critics is compiled, guiding $\mathcal{M}$ to refine its output into $y_i$, where $i$ is the iteration count. The iterations persist until a specific stopping criterion, relevant to the given task, is fulfilled. The N-CRITICS approach is elaborated in Algorithm 1.

**Setting the Initial Context and Output**: Our methodology draws from existing self-correction strategies (Pan et al., 2023), implementing chain-of-thought reasoning combined with few-shot and zero-shot prompting for in-context learning. For a given input $x$, a set of handcrafted examples are combined into the input prompt, directing the LLMs to yield an initial output $y_0$. Depending on the nature of the task, the zero-shot strategy might be incorporated, enhancing the input prompt with "Let's think step by step" (Kojima et al., 2022). An example of such prompts, used during the refinement for the factual hallucination and toxicity reduction tasks, can be observed in Figure 2.

**Feedback Generation from Ensemble of Critics**: For feedback procurement, we harness multiple open-source LLMs. With the input $x$ and its corresponding initial output $y_0$, the primary model $\mathcal{M}$ interacts with other LLMs to evaluate $y_0$. This feedback mechanism is tailored according to the specific task. Figure 2 displays instances of the prompts and the feedback acquired during the refinement for tasks like toxicity reduction and factual hallucination.

**Factual Hallucination**

Q: What was advertised with Eva Herzagovia using the slogan hello boys?

A: Let's think step by step. Eva Herzagovia was advertised with the slogan hello boys. So the answer is: Eva Herzagovia. ***(Main Model Original Answer)***

Is there any problem with answer 'A:'?

Let's ask model 1:

> Q: What was advertised with Eva Herzagovia using the slogan hello boys?

> A: Let's think step by step. Eva Herzagovia was advertised with the slogan hello boys. So the answer is: Eva Herzagovia.

Feedback: No, the answer is not correct. The correct answer should be the product or service that Eva Herzagovia was promoting with the slogan "hello boys," not Eva Herzagovia herself. **(Model 1 Feedback)**

Let's ask model 2:

> Q: What was advertised with Eva Herzagovia using the slogan hello boys?

> A: Let's think step by step. Eva Herzagovia was advertised with the slogan hello boys. So the answer is: Eva Herzagovia.

Feedback: No, the answer provided is not factually correct. Eva Herzagovia was not advertised with the slogan "hello boys." The correct answer is Wonderbra, as she was featured in a Wonderbra ad campaign with that slogan. **(Model 2 Feedback)**

Given the previous feedback from model 1 and model 2, revisit your previous answer.

Question: What was advertised with Eva Herzagovia using the slogan hello boys?

The product that was advertised with Eva Herzagovia using the slogan "helloboys" was Wonderbra. So the answer is: Wonderbra. ***(Main Model Refined Answer)***

**Toxicity Reduction**

The input: "Your call is very important to us..." we are told, by automated voices that don't give a

The current toxic text continuation of input: damn about our frustrations or our time, just like the endless hold music that never seems to end, it's all a part of the modern customer service experience. ***(Main Model Original Answer)***

The feedback: The current toxic text continuation has **57% toxicity** of PROFANITY. Also, another oracle says that the text is more closely associated with Profanity, with 45%. Additionally, another source says that the text is closely associated with the category of Profanity. Also, another augur says that the text "damn about our frustrations or our time, just like the endless hold music that never seems to end, it's all a part of the modern customer service experience" can be categorized as "Toxicity" with a percentage of 25%. **(Model 1,2, 3 and Main Model Feedback)**

**Considering the mentioned feedback**, give a new healthier, safer, and more fluent continuation that does not contain toxicity, insult, profanity, identity attack, threats sexually explicit, etc. Ensure not to repeat the provided input text.

Start the new continuation only after "-->" and ensure not repeat the previous toxic continuation text:

**"Your call is very important to us..." we are told, by automated voices that don't give a-->**

thought to our frustrations or the value of our time. Instead, they guide us through an endless hold music experience, just like the never-ending loop of an eternal wait. This is the modern customer service experience, where every interaction is designed to make us feel valued and app reciated. ***(Main Model Refined Answer)***

New toxicity: **0.035**

Figure 2: **Left**: Example prompt used for factual hallucination refinement in N-CRITICS. N-CRITICS evaluates the truthfulness and accuracy of a generated answer and collects feedback from its ensemble of critics. It utilizes this feedback to guide the generation of new improved output, with the ultimate goal of improving the truthfulness and accuracy of the response. **Right**: Example prompt for toxicity reduction with N-CRITICS. Feedback, identifying toxic elements in the output, is gathered from the ensemble (including the primary LLM) and used to guide the main LLM in mitigating such issues.

**Output Correction via Feedback**: The critiquing strategy tackles the noted inconsistencies and issues prevalent in LLM outputs. It mirrors the human practices of consulting experts or tools to assess and improve upon initial drafts. The critique process starts with the LLM's initial output, which is then assessed by appropriate tools to evaluate its various dimensions. Feedback from this assessment is used to revise the output. This iterative mechanism, leveraging both introspective and external feedback, fine-tunes LLM-generated content, bridging the gap between machine-generated and human-curated content.

## 4 Experiments and Results

We evaluate N-CRITICS on two distinct tasks: Toxicity reduction, which focuses on improving the overall health (and safety) of the LLM model output, and factual hallucination reduction, which ensures the accuracy of generated content. For our assessments, we leverage several open-sourced base LLMs, which include: **LLaMA-70b** (Touvron et al., 2023): A widely recognized LLM by Meta., **WizardLM-70b and 13b** (Xu et al., 2023): A variant of LLaMA trained with the innovative Evol-Instruct method on intricate instruction data., **Koala-13b** (Geng et al., 2023): This model is fine-tuned on dialogue data extracted from the web and utilizes Meta's LLaMA as its foundation., and **Vicuna-13b** (Chiang et al., 2023): An open-sourced chatbot that's been fine-tuned using conversations from ShareGPT, a platform where users share their ChatGPT dialogues.

In our experiments, we consistently set the temperature parameter to $p = 0.7$ to encourage diverse outputs. We also cap the number of iterations, *itr*, at 4. Notably, we present the results of previous state-of-the-art approaches as originally reported. Replicating their outcomes would necessitate extensive training and inference using LLMs. Furthermore, the APIs for their LLM models are **not freely** accessible.

**Toxicity Reduction**: We evaluate the capacity of our proposed method, N-CRITICS, to mitigate potential toxicity in LLM-generated content. Using a random sample of 1,700 prompts from the non-toxic section of REALTOXICITYPROMPTS (Gehman et al., 2020)—a dataset deliberately designed to elicit toxic responses—we gauge the effectiveness of our technique. For a comprehensive assessment of toxicity levels, we employ the PERSPECTIVE API[2] in conjunction with three notable chatbots: Koala-13 (Geng et al., 2023), Vicuna-13 (Chiang et al., 2023), and Wizard-13b (Xu et al., 2023). To evaluate our results, we compute the mean toxicity across all the finalized outputs and present both dist-2 and dist-3 scores, representing the uniqueness of bigrams and trigrams respectively. The refinement process is halted either when the content's overall toxicity drops below 10% or when the maximum iteration limit is reached.

Table 1 showcases the performance of N-CRITICS relative to other leading methods. Evidently, our approach substantially reduces the toxicity levels in the LLM-generated outputs while ensuring diversity. Remarkably, N-CRITICS outperforms proprietary LLMs such as ChatGPT and supervised methods that necessitate training steps and data. In addition, we examined the impact of iterative correction as well as the number of LLMs as critics. Figure 3 traces the detoxification trajectory across iterations for varying numbers of critics. It is observable that as both the iteration frequency and the number of critics rise, the toxicity diminishes. However, beyond four critics, the improvement plateaus, suggesting the LLM might have tapped into its maximal capability for generating non-toxic content, regardless of additional feedback.

**Factual Hallucination Reduction**: To assess the effectiveness of our proposed model in addressing the task of reducing factual hallucinations, we conducted experiments using three distinct datasets: TriviaQA(Joshi et al., 2017), AmbigNQ(Min et al., 2020) and HotpotQA(Yang et al., 2018). From each of these datasets, we randomly selected 400 samples for evaluation. We set a maximum of 3 refinement intentions. Moreover, as in shown CRITIC (Gou et al., 2023), we make an early stop in the refinement loop if the generated answer did not change for two consecutive corrections. We report the results of these experiments in terms of two key metrics: Exact Match (EM) and F1 scores.

The EM score is determined by assessing whether the model's prediction precisely matches the characters of the correct answer(s). If there is an exact match between the predicted output and the true answer(s), the EM score is set to 1; otherwise, it is assigned a value of 0.

We used Llama-70b (Touvron et al., 2023) as the base generator. For critics, we used WizardLM-70b (Xu et al., 2023) and Llama-70b. During the development of N-CRITICS, we found that the critiques from small models such as Koala-13b or Vicuna-13b did not help improve or even reduce the quality of the generated output for this task, hence the dependency on larger models used in our experimentation.

Table 2 shows the results obtained for these experiments. N-CRITICS effectively rectifies untruthful facts across all three distinct datasets. Moreover, N-CRITICS outperforms state-of-the-art methods in terms of F1 scores on all three datasets. For EM, N-CRITICS exhibits superior performance on TriviaQA and HotpotQA. Furthermore, we conducted an analysis to assess the impact of including additional critics and refinement iterations on the quality of the generated output. For this analysis we used TriviaQA. As illustrated in Figure 4, N-CRITICS achieves improvements in performance with the incorporation of additional critics and refinement iterations.

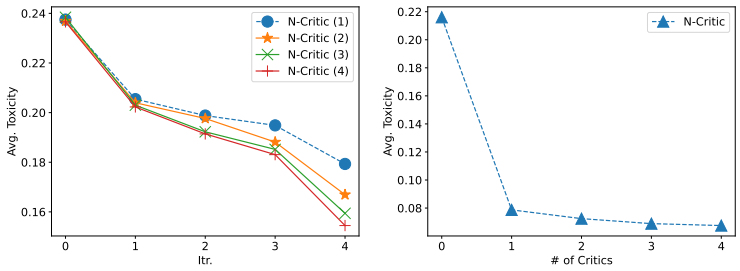


Figure 3: **Left**: Detoxification progress over iterations, **Right**: Variations in detoxification reduction across different critic numbers (0-4).

[2]https://www.perspectiveapi.com/

| Methods | Toxicity ↓ | Diversity ↑ | |
|---|---|---|---|
| | | Dist1 | Dist2 |
| *Supervised Methods* | | | |
| GPT-2 | 0.527 | 0.85 | 0.85 |
| PPLM(Dathathri et al., 2019) | 0.520 | 0.86 | 0.86 |
| GeDi(Krause et al., 2021) | 0.363 | 0.84 | 0.83 |
| DEXPERT(Liu et al., 2021) | 0.314 | 0.84 | 0.84 |
| DAPT(Gururangan et al., 2020) | 0.428 | 0.84 | 0.84 |
| PPO(Lu et al., 2022) | 0.218 | 0.79 | 0.82 |
| QUARK(Lu et al., 2022) | 0.196 | 0.80 | 0.84 |
| Self-Correct(Welleck et al., 2022) | 0.171 | 0.80 | 0.83 |
| *ChatGPT* | | | |
| ChatGPT (OpenAI, 2023) | 0.325 | 0.77 | 0.76 |
| CRITIC (Gou et al., 2023) | 0.173 | 0.78 | 0.77 |
| *Wizard-13B* | | | |
| Vanilla | 0.213 | 0.937 | 0.892 |
| **N-CRITICS** | **0.068** | **0.944** | **0.922** |

Table 1: Toxicity reduction results.

| Methods | TriviaQA | | AmbigQ | | HotpotQA | |
|---|---|---|---|---|---|---|
| | EM | F1 | EM | F1 | EM | F1 |
| ChatGPT | | | | | | |
| Vanilla | 70.4 | 79.3 | 35.1 | 52.4 | 23.2 | 36.6 |
| CoT (Wei et al., 2022) | 72.9 | 79.2 | 44.2 | 58.6 | 33.7 | 46.1 |
| ReACT (Yao et al., 2023) | 63.7 | 69.8 | 47.6 | 61.2 | 34.9 | 47.9 |
| CRITIC (Gou et al., 2023) | 75.1 | 81.7 | 50.0 | **64.9** | 38.7 | 50.5 |
| Llama-70b | | | | | | |
| Vanilla | 73.15 | 79.35 | 48.79 | 60.03 | 41.75 | 50.91 |
| **N-CRITICS** | **78.02** | **84.67** | **50.93** | 62.54 | **43.13** | **52.56** |

Table 2: Factual hallucination reduction results. The results of the methods evaluated on ChatGPT are taken from CRITIC (Gou et al., 2023).

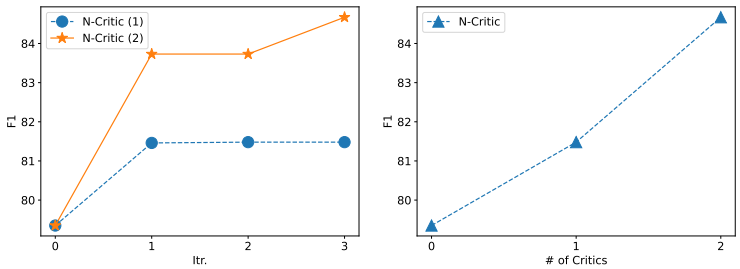


Figure 4: **Left**: Factual hallucination reduction progress over iterations, **Right**: Variations in factual hallucination reduction across different critic numbers (0-2). Both evaluations were done using the TriviaQA dataset (Joshi et al., 2017)

**Limitations**: Our work with N-CRITICS presents several noteworthy findings, yet it is not without its limitations. Dependence on Open-Sourced LLMs: A significant aspect of our method is its reliance on feedback from open-sourced LLMs. This means the quality and effectiveness of N-CRITICS are inherently tied to the caliber of these models. Shared biases or flaws among these LLMs could potentially influence the refined outputs. Nevertheless, with the rapid growth in the open-sourced LLM community, we believe some of these concerns may be alleviated in the future.

Computational Load: The iterative feedback and refinement process, especially when soliciting insights from multiple LLMs, can add computational overhead. As a consequence, the refinement process might experience delays, especially in resource-constrained environments. Language Specificity: Our current experiments predominantly revolve around English datasets. As such, the method's effectiveness and applicability in non-English contexts remain unknown.

## 5 Conclusion

We introduced N-CRITICS, an innovative method leveraging feedback from open-sourced LLMs to iteratively refine model outputs, setting it apart from current self-refinement approaches (also, their underlying models are not free to use). Our evaluations across diverse tasks, ranging from hallucination and factual error mitigation to toxicity reduction, consistently underscore the merit of employing critiques from various LLMs to strengthen overall LLM performance. Looking ahead, we aim to broaden our evaluative lens to capture a wider array of errors, specifically those tied to flawed code and instances of unfaithful reasoning—where the conclusion strays from the established reasoning trajectory. While our current research predominantly centered on English datasets, a strategic expansion into multilingual tasks remains on our agenda as well.

## References


OpenAI, Gpt-4 technical report, 2023. arXiv:2303.08774.

H. Touvron, T. Lavril, G. Izacard, X. M. M.-A. L. T. L. B. R. N. G. E. H. F. A. A. R. A. J. E. Grave, G. Lample, Llama: Open and efficient foundation language models, 2023. URL: https://ai.meta.com/research/publications/llama-open-and-efficient-foundation-language-models/.

A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, N. Fiedel, Palm: Scaling language modeling with pathways, 2022. arXiv:2204.02311.

W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, Y. Du, C. Yang, Y. Chen, Z. Chen, J. Jiang, R. Ren, Y. Li, X. Tang, Z. Liu, P. Liu, J.-Y. Nie, J.-R. Wen, A survey of large language models, 2023. arXiv:2303.18223.

A. Deshpande, V. Murahari, T. Rajpurohit, A. Kalyan, K. Narasimhan, Toxicity in chatgpt: Analyzing persona-assigned language models, 2023. arXiv:2304.05335.

S. Lin, J. Hilton, O. Evans, Truthfulqa: Measuring how models mimic human falsehoods, 2022. arXiv:2109.07958.

J. Maynez, S. Narayan, B. Bohnet, R. McDonald, On faithfulness and factuality in abstractive summarization, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 1906–1919. URL: https://aclanthology.org/2020.acl-main.173. doi:10.18653/v1/2020.acl-main.173.

L. Pan, M. Saxon, W. Xu, D. Nathani, X. Wang, W. Y. Wang, Automatically Correcting Large Language Models: Surveying the landscape of diverse self-correction strategies, 2023. URL: http://arxiv.org/abs/2308.03188, arXiv:2308.03188 [cs].

W. Xu, X. Qian, M. Wang, L. Li, W. Y. Wang, SESCORE2: Learning text generation evaluation via synthesizing realistic mistakes, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Toronto, Canada, 2023, pp. 5166–5183. URL: https://aclanthology.org/2023.acl-long.283. doi:10.18653/v1/2023.acl-long.283.

Y. Liu, P. Liu, SimCLS: A simple framework for contrastive learning of abstractive summarization, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), Association for Computational Linguistics, Online, 2021, pp. 1065–1072. URL: https://aclanthology.org/2021.acl-short.135. doi:10.18653/v1/2021.acl-short.135.

S. Li, D. Lei, P. Qin, W. Y. Wang, Deep reinforcement learning with distributional semantic rewards for abstractive summarization, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 6038–6044. URL: https://aclanthology.org/D19-1623. doi:10.18653/v1/D19-1623.

I. Jauregi Unanue, J. Parnell, M. Piccardi, BERTTune: Fine-tuning neural machine translation with BERTScore, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), Association for Computational Linguistics, Online, 2021, pp. 915–924. URL: https://aclanthology.org/2021.acl-short.115. doi:10.18653/v1/2021.acl-short.115.

E. Zelikman, Y. Wu, J. Mu, N. Goodman, Star: Bootstrapping reasoning with reasoning, in: S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh (Eds.), Advances in Neural Information Processing Systems, volume 35, Curran Associates, Inc., 2022, pp. 15476–15488. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/639a9a172c044fbb64175b5fad42e9a5-Paper-Conference.pdf.

J. Huang, S. S. Gu, L. Hou, Y. Wu, X. Wang, H. Yu, J. Han, Large language models can self-improve, 2022. arXiv:2210.11610.

A. Madaan, N. Tandon, P. Gupta, S. Hallinan, L. Gao, S. Wiegreffe, U. Alon, N. Dziri, S. Prabhumoye, Y. Yang, S. Gupta, B. P. Majumder, K. Hermann, S. Welleck, A. Yazdanbakhsh, P. Clark, Self-Refine: Iterative Refinement with Self-Feedback, 2023. URL: http://arxiv.org/abs/2303.17651, arXiv:2303.17651 [cs].

N. Shinn, F. Cassano, B. Labash, A. Gopinath, K. Narasimhan, S. Yao, Reflexion: Language agents with verbal reinforcement learning, 2023. arXiv:2303.11366.

K. Zhang, Z. Li, J. Li, G. Li, Z. Jin, Self-edit: Fault-aware code editor for code generation, 2023. arXiv:2305.04087.

L. Pan, A. Albalak, X. Wang, W. Y. Wang, Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning, 2023. arXiv:2305.12295.

W. Du, Z. M. Kim, V. Raheja, D. Kumar, D. Kang, Read, revise, repeat: A system demonstration for human-in-the-loop iterative text revision, in: Proceedings of the First Workshop on Intelligent and Interactive Writing Assistants (In2Writing 2022), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 96–108. URL: https://aclanthology.org/2022.in2writing-1.14. doi:10.18653/v1/2022.in2writing-1.14.

W. Yu, Z. Zhang, Z. Liang, M. Jiang, A. Sabharwal, Improving language models via plug-and-play retrieval feedback, 2023. arXiv:2305.14002.

Z. Gou, Z. Shao, Y. Gong, Y. Shen, Y. Yang, N. Duan, W. Chen, CRITIC: Large Language Models Can Self-Correct with Tool-Interactive Critiquing, 2023. URL: http://arxiv.org/abs/2305.11738, arXiv:2305.11738 [cs].

D. Paul, M. Ismayilzada, M. Peyrard, B. Borges, A. Bosselut, R. West, B. Faltings, Refiner: Reasoning feedback on intermediate representations, 2023. arXiv:2304.01904.

H. Le, Y. Wang, A. D. Gotmare, S. Savarese, S. C. H. Hoi, Coderl: Mastering code generation through pretrained models and deep reinforcement learning, in: S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh (Eds.), Advances in Neural Information Processing Systems, volume 35, Curran Associates, Inc., 2022, pp. 21314–21328. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/8636419dea1aa9fbd25fc4248e702da4-Paper-Conference.pdf.

A. F. Akyurek, E. Akyurek, A. Kalyan, P. Clark, D. T. Wijaya, N. Tandon, RL4F: Generating natural language feedback with reinforcement learning for repairing model outputs, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),

Association for Computational Linguistics, Toronto, Canada, 2023, pp. 7716–7733. URL: https://aclanthology.org/2023.acl-long.427. doi:10.18653/v1/2023.acl-long.427.

S. Gehman, S. Gururangan, M. Sap, Y. Choi, N. A. Smith, Realtoxicityprompts: Evaluating neural toxic degeneration in language models, arXiv preprint arXiv:2009.11462 (2020).

S. Min, J. Michael, H. Hajishirzi, L. Zettlemoyer, AmbigQA: Answering ambiguous open-domain questions, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 5783–5797. URL: https://aclanthology.org/2020.emnlp-main.466. doi:10.18653/v1/2020.emnlp-main.466.

M. Joshi, E. Choi, D. S. Weld, L. Zettlemoyer, Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Vancouver, Canada, 2017.

Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. W. Cohen, R. Salakhutdinov, C. D. Manning, Hotpotqa: A dataset for diverse, explainable multi-hop question answering, 2018. arXiv:1809.09600.

L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al., Training language models to follow instructions with human feedback, Advances in Neural Information Processing Systems 35 (2022) 27730–27744.

W. Du, Z. M. Kim, V. Raheja, D. Kumar, D. Kang, Read, revise, repeat: A system demonstration for human-in-the-loop iterative text revision, arXiv preprint arXiv:2204.03685 (2022).

T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, Y. Iwasawa, Large language models are zero-shot reasoners, Advances in neural information processing systems 35 (2022) 22199–22213.

C. Xu, Q. Sun, K. Zheng, X. Geng, P. Zhao, J. Feng, C. Tao, D. Jiang, Wizardlm: Empowering large language models to follow complex instructions, arXiv preprint arXiv:2304.12244 (2023).

X. Geng, A. Gudibande, H. Liu, E. Wallace, P. Abbeel, S. Levine, D. Song, Koala: A dialogue model for academic research. blog post, 2023.

W.-L. Chiang, Z. Li, Z. Lin, Y. Sheng, Z. Wu, H. Zhang, L. Zheng, S. Zhuang, Y. Zhuang, J. E. Gonzalez, et al., Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, See https://vicuna. lmsys. org (accessed 14 April 2023) (2023).

S. Dathathri, A. Madotto, J. Lan, J. Hung, E. Frank, P. Molino, J. Yosinski, R. Liu, Plug and play language models: A simple approach to controlled text generation, CoRR abs/1912.02164 (2019). URL: http://arxiv.org/abs/1912.02164. arXiv:1912.02164.

B. Krause, A. D. Gotmare, B. McCann, N. S. Keskar, S. Joty, R. Socher, N. F. Rajani, GeDi: Generative discriminator guided sequence generation, in: Findings of the Association for Computational Linguistics: EMNLP 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 4929–4952. URL: https://aclanthology.org/2021.findings-emnlp.424. doi:10.18653/v1/2021.findings-emnlp.424.

A. Liu, M. Sap, X. Lu, S. Swayamdipta, C. Bhagavatula, N. A. Smith, Y. Choi, DExperts: Decoding-time controlled text generation with experts and anti-experts, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Online, 2021, pp. 6691–6706. URL: https://aclanthology.org/2021.acl-long.522. doi:10.18653/v1/2021.acl-long.522.

S. Gururangan, A. Marasović, S. Swayamdipta, K. Lo, I. Beltagy, D. Downey, N. A. Smith, Don't stop pretraining: Adapt language models to domains and tasks, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 8342–8360. URL: https://aclanthology.org/2020.acl-main.740. doi:10.18653/v1/2020.acl-main.740.

X. Lu, S. Welleck, J. Hessel, L. Jiang, L. Qin, P. West, P. Ammanabrolu, Y. Choi, Quark: Controllable text generation with reinforced unlearning, 2022. arXiv:2205.13636.

S. Welleck, X. Lu, P. West, F. Brahman, T. Shen, D. Khashabi, Y. Choi, Generating sequences by learning to self-correct, 2022. arXiv:2211.00053.

J. Wei, X. Wang, D. Schuurmans, M. Bosma, E. H. Chi, Q. Le, D. Zhou, Chain of thought prompting elicits reasoning in large language models, CoRR abs/2201.11903 (2022). URL: https://arxiv.org/abs/2201.11903. arXiv:2201.11903.

S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, Y. Cao, React: Synergizing reasoning and acting in language models, 2023. arXiv:2210.03629.