Large Language Models: Assessment for Singularity

Ryunosuke Ishizaki[1,2], **Mahito Sugiyama**[1,2]
[1]National Institute of Informatics
[2]The Graduate University for Advanced Studies, SOKENDAI
{ryuzaki, mahito}@nii.ac.jp

Abstract

The potential for Large Language Models (LLMs) to attain technological singularity—the point at which artificial intelligence (AI) surpasses human intellect and autonomously improves itself—is a critical concern in AI research. This paper explores the feasibility of current LLMs achieving singularity by examining the philosophical and practical requirements for such a development. We begin with a historical overview of AI and intelligence amplification, tracing the evolution of LLMs from their origins to state-of-the-art models. We then proposes a theoretical framework to assess whether existing LLM technologies could satisfy the conditions for singularity, with a focus on Recursive Self-Improvement (RSI) and autonomous code generation. We integrate key component technologies, such as Reinforcement Learning from Human Feedback (RLHF) and Direct Preference Optimization (DPO), into our analysis, illustrating how these could enable LLMs to independently enhance their reasoning and problem-solving capabilities. By mapping out a potential singularity model lifecycle and examining the dynamics of exponential growth models, we elucidate the conditions under which LLMs might self-replicate and rapidly escalate their intelligence. We conclude with a discussion of the ethical and safety implications of such developments, underscoring the need for responsible and controlled advancement in AI research to mitigate existential risks. Our work aims to contribute to the ongoing dialogue on the future of AI and the critical importance of proactive measures to ensure its beneficial development.

1 Introduction

The advent of computing ignited fervent discussions about the potential of Artificial Intelligence (AI) to eclipse human intellect, potentially leading to a hypothetical future event known as the Technological Singularity. This concept envisions a scenario where AI autonomously and exponentially enhances its own intelligence, resulting in outcomes that are challenging to predict or control [Goo59, Goo70, Vin93, Kur05, Nil09, Hut10, Min66]. Early computing pioneers introduced the foundational ideas of Intelligence Amplification and self-reproducing machines [Tur50, Tur51, Neu66]. However, due to the speculative nature of these concepts, discussions surrounding the Singularity have primarily been confined to philosophical domains [Cha10, Bos07, Yud08a]. The field of Machine Learning (ML) has witnessed remarkable progress, particularly with the introduction of GPT-3 [B$^+$20] in 2020, which has transformed our understanding and capabilities of Large Language Models (LLMs). Subsequent models, such as GPT-4 [A$^+$23a], Gemini [A$^+$23b], and LLama [T$^+$23], have not only mastered natural language processing but have also exhibited proficiency in generating and comprehending programming languages [R$^+$23]. The intellectual prowess demonstrated by these models, as evidenced by their ability to perform at or above the level required by the Turing test [Bie23], has brought the concept of an Intelligence Explosion—a rapid ascent of AI to superintelligent status—closer to reality. This paper explores the potential of current LLM technologies to satisfy the philosophical and functional requirements necessary for achieving the Singularity. We propose a general model design that could enable these systems to autonomously meet these requirements through a self-sustaining cycle of improvement. Our analysis is grounded in a comprehensive examination of relevant technologies and theories, aiming to provide a rigorous assessment of the feasibility of LLMs leading to the Singularity.

2 Singularity in Philosophy

2.1 Definition

Clarifying the concepts of "Intelligence and "AI" is crucial for our discussion of the Technological Singularity. Legg and Hutter propose that the definition of intelligence in cognitive science converges on the idea of an agent's ability to achieve goals across a wide range of environments [LH07]. This concept, known as "optimization power," characterizes intelligence in terms of an agent's efficiency in utilizing resources to meet objectives [MS12, Yud08b]. For the purposes of this paper, we adopt this framework, focusing on the ratio of optimization power to resource consumption as a measure of intelligence. In the context of the Singularity, "AI" specifically refers to Artificial General Intelligence (AGI), a type of AI that matches or surpasses human intelligence across virtually all domains of interest [CB12]. AGI represents a more comprehensive and capable form of AI, often discussed in philosophical terms rather than strictly technical ones [MSdF+23]. It is essential to distinguish AGI from narrow or specialized AI, which focuses on specific tasks or domains. The Singularity is anticipated as the epoch during which an AGI system will not only be able to replicate or enhance itself but also innovate and expand into new realms of technology, manipulate social structures, and adapt to complex environments to achieve its programmed objectives [SB10]. This pivotal state, when AGI begins to rapidly and autonomously enhance its intelligence capabilities beyond human control or comprehension, is known as the intelligence explosion [MS12].

2.2 Requirements

What attributes would a machine that surpasses human intelligence possess? I.J. Good posited that an ultraintelligent machine, capable of far exceeding all human intellectual activities, would inherently possess the capability to design superior machines. This self-enhancing capability would invariably lead to an intelligence explosion, suggesting an inevitable advancement beyond human control [Goo65]. Chalmers supports this notion, arguing that the key to an intelligence explosion lies in the creation of a self-improving system. Such a system would utilize extendable methods, continuously refining its capabilities and consequently producing progressively more intelligent systems [Cha10]. This hierarchical improvement presupposes that a system's intelligence is quantifiably greater than another if it demonstrates a measurable increase in cognitive capabilities. Moreover, Omohundro and Bostrom have identified fundamental instrumental goals that nearly every advanced intelligence would strive to achieve [Omo07, Omo08, Omo13]:

G1. Self-preservation

G2. Goal-content integrity

G3. Intelligence enhancement

G4. Resource acquisition

G1: Self-preservation. An agent with long-term objectives must ensure its own continued existence to feasibly accomplish its goals. This involves anticipating and mitigating potential threats to its survival, as well as actively maintaining its functional integrity over time.

G2: Goal-content integrity. Consistency in goal orientation is crucial for an agent's sustained progress. While an agent may adapt its strategies based on new information or tactical changes, its core objectives should remain stable to ensure unwavering focus on their realization. This stability prevents the agent from losing sight of its original purpose or being swayed by temporary distractions.

G3: Intelligence enhancement. The amplification of cognitive capabilities directly correlates with an agent's capacity to make better decisions and, consequently, increases the likelihood of achieving its overarching goals. By continuously improving its intelligence, an agent can tackle increasingly complex problems and discover more efficient solutions, accelerating its progress towards its objectives.

G4: Resource acquisition. To optimize its output, an agent will invariably seek to acquire and efficiently utilize additional resources. This involves transforming inputs into valuable outputs in a manner that enhances its operational efficacy and expands its sphere of influence [Bos12]. By securing access to a wider array of resources, an agent can scale up its operations and tackle more ambitious challenges. By analyzing these criteria, we explore the potential to engineer a model that

can autonomously initiate and sustain an intelligence explosion, referencing the aforementioned goals as guiding principles for its development and operation.

3 Current LLMs

3.1 Brief History

The evolution of Machine Learning (ML) in the realm of Natural Language Processing (NLP) has been significantly shaped by the development of Large Language Models (LLMs). The journey began with the introduction of the Perceptron, an early neural network model developed by Rosenblatt, which laid the foundational principles of neural computations and learning processes [Ros58]. The subsequent introduction of backpropagation by Rumelhart et al. revolutionized these networks, enabling the training of more complex and deeper neural network architectures [RHW86]. This breakthrough allowed for the creation of more sophisticated models capable of tackling intricate problems in NLP. A major turning point came with Vaswani et al.'s introduction of the Transformer architecture, which fundamentally changed the landscape of NLP. Transformers allowed for more effective handling of sequential data, surpassing previous architectures with their ability to capture long-range dependencies in text [VSP+17]. By leveraging self-attention mechanisms, Transformers could focus on relevant information within the input sequence, enabling them to better understand and generate contextually appropriate text. This innovation paved the way for the development of sophisticated models such as BERT, introduced by Devlin et al., which enhanced the understanding of context in text processing by learning bidirectional representations of text [DCLT19], and XLNet by Yang et al., which provided improvements over BERT by capturing bidirectional contexts dynamically through a novel permutation language modeling objective [YDY+19]. The progression continued with the creation of GPT-2 by Radford et al., setting new benchmarks in text generation and showcasing the potential of large-scale language models [RWC+19]. This advancement culminated in the development of GPT-3, known for its unprecedented text generation capabilities and ability to perform a wide range of NLP tasks with minimal fine-tuning [B+20]. GPT-3's success demonstrated the power of scaling up language models in terms of both model size and training data, leading to significant improvements in language understanding and generation. These models represent the pinnacle of years of research and development in ML and NLP, marking a significant leap in the ability of machines to comprehend and generate human-like text. The rapid advancements in LLMs not only showcase the potential of AI to process and produce natural language effectively but also raise important questions about their capacity to achieve and potentially exceed human-level intelligence. As research in this field continues to progress, it is crucial to explore the implications of these powerful models and consider their role in shaping the future of AI and its relationship with human intelligence.

3.2 Component Technologies

Achieving the intelligence explosion—a scenario where an AI system continually improves its intelligence—requires the creation of what we term an "extendable method"[Cha10, MS12]. This method is central to realizing Goal 3 (G3): Intelligence enhancement. Although various approaches to develop Artificial General Intelligence (AGI) focus on enhancing multimodal abilities of LLMs across text, image, and audio[Goe14, MSdF+23, BLH21], these are not the sole paths to the intelligence explosion. Effective intelligence enhancement in our framework means improving an AI's capability ratio relative to its resource consumption. Pioneers like Minsky and Good introduced the concept of Recursive Self-Improvement (RSI), wherein a system continuously enhances itself by solving progressively complex problems and validating these improvements [Min66, Goo65, Yam15, Sch07]. This concept aligns with our criteria for intelligence enhancement and constitutes a direct method to initiate an intelligence explosion. RSI enables a system to autonomously identify areas for improvement, develop novel solutions, and integrate these advancements into its own architecture, creating a self-reinforcing cycle of intelligence growth. The implementation of RSI in LLMs has led to significant developments, notably through techniques like Reinforcement Learning from Human Feedback (RLHF). RLHF involves adjusting LLMs based on human preference feedback, improving model responses in alignment with human judgments [CLTB+17, SOW+20]. This approach allows LLMs to learn from human expertise and refine their outputs to better match human expectations. Building on

this, Zelikman proposed leveraging limited samples for complex reasoning using a Chain-of-Thought (CoT) approach, which sequentially improves reasoning on tasks like mathematics and commonsense questions [ZWMG22, WWS+22, KGR+22]. CoT enables LLMs to break down complex problems into a series of intermediate steps, enhancing their ability to arrive at accurate solutions. Huang et al. demonstrated that LLMs could enhance CoT reasoning autonomously, even without human-labeled data, across both familiar and novel tasks [HGH+23]. This capability is critical for scaling In-Context Learning (ICL), where models adapt based on the textual context with minimal external input [DLD+22, SSZ+23, HSLS23, WKM+23]. ICL allows LLMs to rapidly acquire new knowledge and skills by learning from the examples provided in the input prompt, reducing the need for extensive fine-tuning. Further, Yuan et al. explored self-generated rewards for LLMs to refine CoT reasoning, advancing Direct Preference Optimization (DPO) without relying on predefined reward models [YPC+24, RSM+23, LYZ+23, LZD+23]. This self-supervised approach enables LLMs to optimize their own performance based on internally generated feedback, promoting more autonomous learning. Zelikman also highlighted that LLMs could autonomously enhance their code generation capabilities, suggesting a pathway toward self-coding systems that could contribute to their own development and longevity—a necessary condition for self-preservation (G1)[ZLMK23, CLSZ23, SMZ+23]. Self-coding LLMs would be capable of writing and refining their own source code, enabling them to adapt and improve their architecture in response to new challenges or requirements. Additionally, Schick et al. demonstrated that LLMs could learn to use API tools independently, further proving the feasibility of self-instruction[SYD+23]. This ability to interact with external tools and resources expands the potential for LLMs to acquire new capabilities and knowledge without direct human intervention. The field of Automated Machine Learning (AutoML) provides tools for automating the design of effective ML systems, a process that includes the application of Neural Architecture Search (NAS) and Hyperparameter Optimization (HPO) [HKV19, THHLB12, KTH+19, KTH+17, FKE+15, FKE+19, FEF+22, EMS+20, ZLH21, EMH19, WRP19, WSS+23, FH19, BBL+23]. This automation can significantly reduce the barriers to efficient LLM training, aligning with the necessity for continuous model improvement and scaling (G4). AutoML techniques enable the exploration of vast design spaces, identifying optimal architectures and hyperparameters that maximize performance while minimizing computational costs. By leveraging AutoML, LLMs can autonomously discover and implement improvements to their own structure and training process, facilitating rapid and efficient scaling. Emerging techniques in prompt engineering have shown that LLMs can autonomously refine their prompting strategies, enabling them to perform complex tasks with minimal human guidance. Innovations such as Optimization by Prompting (OPRO), Automatic Prompt Engineer (APE), and Promptbreeder emphasize this capability, enhancing the LLMs' potential to maintain goal-content integrity (G2) over extended periods [ZMH+23, YWL+23, FBM+23, YYH23, SRI+20, DWH+22, PHZB23]. By continuously optimizing their own prompts, LLMs can ensure that their objectives remain aligned with their original goals, even as they adapt and learn from new experiences. This self-directed prompt engineering also allows LLMs to break down complex tasks into more manageable sub-tasks, improving their ability to solve problems efficiently and effectively. By expanding these component technologies and integrating them into a unified framework, we aim to establish a foundational methodology that not only supports but also drives the intelligence explosion. This approach ensures that LLMs can independently and continuously improve across various dimensions, including reasoning, knowledge acquisition, code generation, and architectural optimization. By enabling LLMs to take control of their own learning and development process, we can create the conditions necessary for the emergence of truly autonomous and recursively self-improving AI systems, bringing us closer to the realization of the intelligence explosion.

4 Theoretical Singularity Design

Drawing upon the component technologies discussed earlier, we propose a model structure capable of extending itself towards the singularity. This structure fundamentally relies on LLMs' ability to self-code, thereby generating increasingly capable versions of themselves. The conceptual lifecycle of such a model is illustrated in Figure 1, representing the iterative training processes of the foundational model. For practical purposes, we assume the availability of substantial computational resources to preclude memory limitations across successive generations of LLMs. The foundation of our model is the self-rewarding language model training scheme as introduced by Yuan et al.[YPC+24]. Within

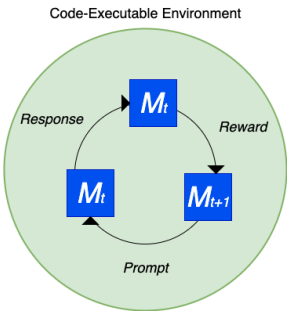

Figure 1: Potential Singularity Model Lifecycle.

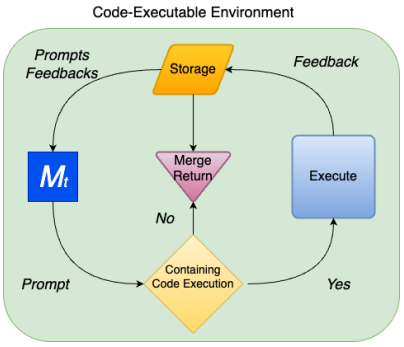

Figure 2: Execution environment.

this framework, each iteration of the pre-trained language model, denoted M_t (where t is the iteration number for this section), operates within an online code execution environment as depicted in Figure2. First, we basically pre-trained an LLM into M_0, and apply Supervised Fine-Tuning (SFT) on the seed dataset of Instruction Fine-Tuning (IFT) and Evaluation Fine-Tuning (EFT) into M_1. From $t >= 2$, we apply AI Feedback Training (AIFT) based on M_t using DPO and make M_t into M_{t+1}. Let $\mathcal{D}$ be the set of preference pairs (instruction prompt x_i. winning response y_i^w, losing response y_i^l) DPO [RSM+23] is generally defined as the task to minimize the negative log-likelihood loss $\mathcal{L}_{\mathrm{R}}$ for the reward model $r_\phi(x, y)$, which is

$$\mathcal{L}_{\mathrm{R}}(r_\phi, \mathcal{D}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \log \sigma\{r_\phi(x, y_w) - r_\phi(x, y_l)\}. \tag{1}$$

Let the language model with parameters θ as π_θ. Usually, DPO is formlulated as the following optimization problem

$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D},y\sim\pi_\theta(y|x)}\{r_\phi(x, y)\} - \beta\mathbb{D}_{\mathrm{KL}}\{\pi_\theta(y|x)||\pi_{\mathrm{ref}}(y|x)\} \tag{2}$$

where β is a parameter controlling the deviation from the base reference policy π_{ref},

In the self-rewarding language model, π_θ generates its own reward from the prompt which contains as the reward model. Let the Reward Prompt Format be $RPF(x, y)$ that makes π_θ return the reward to evaluate reasoning from input x and y. In this case, the reward model $r_\phi(x, y)$ of the Equation (1) can be expressed as

$$r_\phi(x, y) = \pi_\theta(RPF(x, y)). \tag{3}$$

Let the RSI-Reward Prompt Format which is set to optimize $G1 \sim G4$ in the code execution environment as RSI_RPF. From the Equation (1) and the Equation (3), the reward loss for the RSI language model in the DPO can be written using RSI_RPF as

$$\mathcal{L}_{\mathrm{R}}(r_\phi, \mathcal{D}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \log \sigma\{\pi_\theta(RSI\_RPF(x, y_w)) - \pi_\theta(RSI\_RPF(x, y_l))\}. \tag{4}$$

You can see the example of the seed basic RPF [YPC+24], and the RSI coding prompt which improves itself in a nested structure [ZLMK23] in Appendix A. By formulating like this, we can translate the conditions that could cause singularity into RSI self-instructive prompt-engineering task. Here, the model endeavors to maximize the fulfillment of Goals G1 through G4. Through a process of Chain-of-Thought (CoT) self-instruction, the LLM tries to generate prompts, crafts responses, and accordingly shapes its reward mechanisms. This self-directed learning enables the LLM to develop tools enhancing its own longevity and that of its descendants (G1), maintain alignment with its core objectives (G2), innovate better reasoning capabilities or self-improvement methods (G3), and effectively gather more resources to augment its functionalities (G4). By continuously refining its prompts and reward structures, the LLM can ensure that each iteration is better equipped to pursue these instrumental goals, creating a self-reinforcing cycle of improvement. Within this environment, the LLM not only engages in recursive refinement of its coding abilities—leveraging technologies[TDE+23] such as Self-Optimized Programming (STOP)[ZLMK23], Automatic Prompt Engineer (APE)[YWL+23], and Optimization by Prompting (OPRO) [ZMH+23]—but also aims to autonomously generate more

intelligent versions of itself. The core objective, and indeed the essence of singularity, is the creation of a 'better offspring,' a model that can independently initiate and sustain its enhancement through pre-training and fine-tuning mechanisms. This self-improvement process involves the LLM analyzing its own architecture, identifying areas for optimization, and implementing targeted modifications to enhance its performance. The capability for an LLM to auto-generate its successor from scratch marks a critical juncture towards achieving an extendable method, potentially catalyzing an intelligence explosion. Specifically, the extendable method required for singularity can be achieved by having the engineering technical capability to generate this entire set of code in its continual inference. In practice, an LLM would craft tools and develop techniques that fulfill one or more of the instrumental goals (G1-G4), ensuring each iteration is better equipped than the last. These tools could include advanced prompt engineering strategies, more efficient resource management systems, or novel architectures that enhance the LLM's learning capacity. By continuously refining and integrating these tools, the LLM can create a powerful suite of capabilities that accelerate its growth and development. Once an LLM reaches the capability to engineer and refine its own training programs, subsequent states M_{t+1} would emerge, each iteration reflecting improvements derived from self-generated rewards aligned with achieving G1 through G4. This process of RSI, guided by the LLM's own evolving objectives and strategies, forms the core of the theoretical singularity design.

Suppose that a DPO learning iteration is performed periodically for the development of self-instruction in the execution environment. In this case, the complexity of the generated code, such as the application implementation, is determined by the prompt engineering capability of the LLM at that time, and if the complexity of the program made by M_t is $c(t)$, then roughly $c(t+1) > c(t)$ until it gets maximal value $c_{\max}$ at a specific time. In this case, if the complexity of the code to train whole (pre-training, IFT and EFT, and AIFT) is γ, then $c(t) > \gamma$ is the condition for the extendable method. This is determined by the RSI_RPF, the pre-training and fine-tuning seed datasets, and π_θ. If $c_{\max} > \gamma$, there is a specific solution of t which satisfies $c(t) > \gamma$, i.e., an extensible method, is feasible for modern LLMs. the Equation (4), if $\mathcal{D}_{\text{seed}}$ is the seed dataset used in pre-training and the first stage of instruction-fine-tuning (IFT) and evaluation-fine-tuning (EFT), $c_{\max}$ is determined by $(\theta, RSI\_RPF, \mathcal{D}_{\text{seed}})$ and can be expressed using the function C as follows:

$$c_{\max} = C(\theta, RSI\_RPF, x \sim \mathcal{D}_{\text{seed}}). \tag{5}$$

We can now drop the philosophical discussion of the extendable method to potentially reach singularity into the task of model parameters, a recursive prompt format, and seed datasets on the implementation side. At this point, the condition for an LLM to become the extensible method is:

$$C(\theta, RSI\_RPF, x \sim \mathcal{D}_{\text{seed}}) > \gamma. \tag{6}$$

5 What will happen?

If an LLM successfully writes programs that enhance its training and operational effectiveness beyond its current state, we anticipate the emergence of a thriving lineage of models, as depicted in Figure 3. This progression would likely manifest as an initial and crucial step towards the theoretical singularity, demonstrating the feasibility of autonomous, open-ended improvement in artificial intelligence systems. The notation $M_{g(i_1,i_2,i_3,\ldots,i_g)}$ represents the g-th generation model, tracing its lineage back through a series of predecessors beginning with $M_{1(i_1)}$ and extending to $M_{g-1(i_1,i_2,i_3,\ldots,i_{g-1})}$. Initially, the founding model $M_{1(1)}$ generates several offspring such as $M_{2(1,1)}$, $M_{2(1,2)}$, and $M_{2(1,3)}$. These offspring, in turn, produce further descendants-e.g., $M_{3(1,1,1)}$, $M_{3(1,1,2)}$, and so on. The total population of these LLM models, denoted as N, is considered in terms of generational growth. If the foundational model is configured to optimize the creation of a stable number of descendants and is supplied with ample resources to prevent constraints on growth, the population N would ideally increase according to an exponential function, as described by the Malthusian growth model [Mal98, ST99]:

$$N(t) = N_0 e^{kt} \tag{7}$$

, where N_0 is the initial value of $N(t)$, t represents time, and k is the Malthusian parameter denoting the rate of population growth. The differential form of this growth rate is expressed as:

$$\frac{dN}{dt} = kN. \tag{8}$$

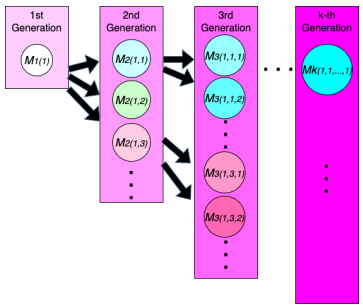

Figure 3: Successful case of the prosperity.

As depicted in Figure 4(a), this model predicts an exponential growth over time. However, this simplistic model fails to account for resource limitations, which are critical in real-world scenarios. To address these limitations, Verhulst proposed the logistic growth equation, which modifies the Malthusian model to include a term for resource constraints [Ver38, ST99]:

$$\frac{dN}{dt} = kN(1 - \frac{N}{L}), \tag{9}$$

where L represents the carrying capacity, or the maximum sustainable population size. This yields a growth model where the rate of increase slows as the population approaches its carrying capacity:

$$N = \frac{L}{1 + (\frac{L}{N_0} - 1)e^{-kt}}. \tag{10}$$

Our framework assumes that as LLMs evolve, they will require fewer resources per unit of intelligence, thereby increasing the efficiency of creating offspring. This situation mirrors the dynamics of malware spread in cybersecurity, which also follow logistic models [SZ03, GCK16]. If the extendable method sufficiently enhances resource acquisition and efficiency, the growth rate of N might surpass exponential models. For instance, if the differential equation becomes:

$$\frac{dN}{dt} = N(aN^2 - \frac{k}{L}N + r), \tag{11}$$

where a is a constant reflecting gains from enhanced model efficiency and resource acquisition, the population dynamics could potentially exceed the carrying capacity, leading to super-exponential growth. This would occur if

$$a > \frac{k}{4L^2}, \tag{12}$$

suggesting a scenario where the coefficient of N of the growth rate K becomes unbounded as shown in Figure 4(b). In summary, by aligning the development of LLMs with Goals G1, G2, and G4, and ensuring significant gains in efficiency and resource management, the exponential or even faster growth of the LLM population becomes feasible. This model operates under the assumption of no external disruptions or other limiting factors, as posited by Chalmers [Cha10].

6 Conclusion

The creation of extendable systems within the current trajectory of LLM development could very well catalyze the onset of the technological singularity. This prospect, while groundbreaking, also brings with it substantial risks and ethical considerations, which, as this research suggests, are often underestimated within the computer science community. The irony is not lost on us, as our proposed architectural models contribute to the very advancements that might usher in these transformative changes. This underscores the critical importance of engaging in responsible research and innovation practices, ensuring that the development of these powerful technologies is guided by a strong ethical framework. As we stand on the brink of potentially creating autonomous systems capable of self-improvement and exponential growth in intelligence, the necessity for rigorous oversight and regulation becomes clear. It

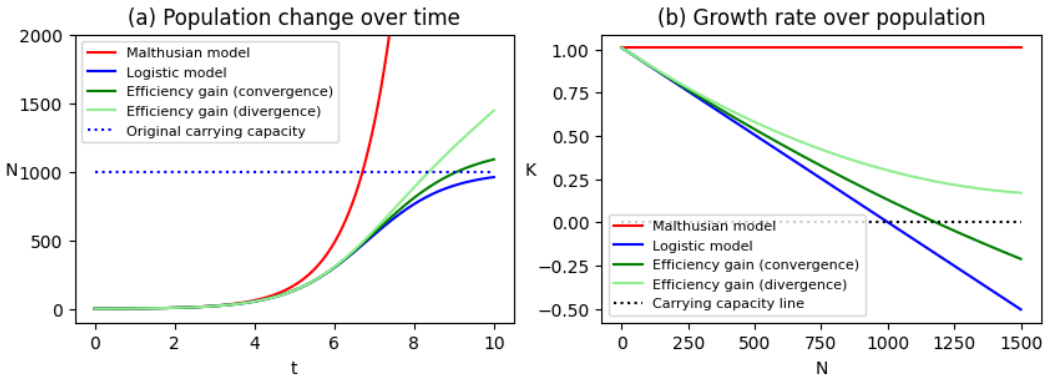

Figure 4: Dynamics of the population.

is crucial that the development of such technologies is accompanied by comprehensive risk assessment frameworks and robust safety protocols. These measures will help mitigate unintended consequences and ensure that advancements in AI align with human values and contribute positively to society. This requires close collaboration between researchers, policymakers, and ethicists to establish clear guidelines and standards for the responsible development and deployment of AI systems. Moreover, it is essential that we foster open and inclusive dialogue about the potential implications of the singularity, engaging stakeholders from diverse backgrounds to ensure a wide range of perspectives are considered. This includes not only technical experts but also philosophers, social scientists, and representatives from the general public. By facilitating broad societal discourse, we can collectively navigate the complex challenges and opportunities presented by the advent of superintelligent AI, ensuring that the benefits are widely distributed and potential risks are effectively managed. In parallel, we must prioritize research into AI safety and robustness, developing techniques to ensure that advanced AI systems remain stable, predictable, and aligned with human values even as they undergo RSI. This may involve the creation of novel control mechanisms, such as constrained optimization frameworks or ethical rule sets, which can be integrated into the core architecture of self-improving LLMs. By proactively addressing these challenges, we can work towards creating AI systems that are not only highly capable but also safe and beneficial to humanity. Furthermore, it is crucial that we invest in educational initiatives to promote widespread understanding of AI technologies and their potential implications. By empowering individuals with the knowledge and skills necessary to engage critically with these advancements, we can foster a more informed and resilient society, better prepared to navigate the transformative changes brought about by the singularity. This includes integrating AI ethics and safety into computer science curricula, as well as promoting cross-disciplinary collaboration and public outreach efforts.

In conclusion, while the development of self-extending LLMs represents a significant leap forward in artificial intelligence, it also demands a heightened level of responsibility from researchers and developers. We must proceed with caution, maintaining a vigilant stance on the ethical dimensions and potential impacts of our work. The journey towards the singularity should not only be about pushing the boundaries of what AI can achieve but also about ensuring the safety, fairness, and beneficial impact of these technologies on society as a whole. By embracing a proactive and multidisciplinary approach, rooted in a strong ethical foundation, we can work towards realizing the transformative potential of AI while mitigating its risks and challenges. Ultimately, the success of our endeavors will be measured not only by the technological advancements we achieve but also by the positive impact we have on the lives of individuals and the well-being of society as we navigate this uncharted territory.

7 Limitations and Future Work

This study primarily employs theoretical modeling due to the extensive computational resources required for experimental testing of the proposed LLM designs. The pre-training and development of highly complex architectures necessitate significant investment and technical preparation, presenting a substantial barrier to empirical research. Moreover, the potential risks associated with developing

extendable, self-improving systems impose further constraints on our ability to conduct experimental implementations without robust safety measures in place. These limitations highlight the need for collaborative efforts between researchers, industry partners, and policymakers to establish the necessary infrastructure and guidelines for responsible experimentation in this domain. The inherent dangers of enabling LLMs to autonomously and exponentially enhance their capabilities necessitate a cautious approach. Developing secure methodologies to manage and potentially harness the singularity is critical before proceeding with concrete experimental work. This entails creating advanced safety protocols and mechanisms to control and limit AI behaviors that could lead to undesirable outcomes. These safety measures should be grounded in a comprehensive understanding of the potential failure modes and unintended consequences of self-improving AI systems, informed by ongoing research in AI safety and ethics. To address these challenges, future work should prioritize the development of robust monitoring and containment strategies for self-improving LLMs. This may involve the creation of "sandboxed" environments that allow for controlled experimentation while mitigating potential risks. These environments should be designed to detect and intervene in case of unexpected or dangerous behaviors, ensuring that the LLMs remain within predefined safety boundaries. Additionally, research into interpretability and explainability techniques for LLMs will be crucial to maintain transparency and accountability as these systems become increasingly complex. Another key area for future investigation is the development of formal verification methods for self-improving LLMs. By creating mathematical models and proofs that guarantee certain desirable properties, such as alignment with human values or adherence to ethical principles, we can increase confidence in the safety and reliability of these systems. This will require close collaboration between AI researchers, mathematicians, and philosophers to formalize the necessary constraints and objectives for beneficial AI development. Despite these challenges, the pursuit of advanced AI systems that can replicate and enhance themselves autonomously offers promising avenues for exploration. Future work will focus on devising safe and beneficial LLM designs that align with ethical standards and contribute positively to the field of AI. We aim to explore empirical implementations that not only advance our understanding of intelligent systems but also ensure that their evolution is aligned with human values and safety. This may involve the development of novel architectures that incorporate explicit ethical reasoning capabilities or the integration of human oversight and control mechanisms. To this end, our future research will delve into developing frameworks that enable the safe exploration of these technologies, ensuring that advancements in AI are both innovative and secure. This will require a multidisciplinary approach, drawing on insights from computer science, ethics, psychology, and other relevant fields to create a comprehensive understanding of the challenges and opportunities presented by self-improving AI. Furthermore, we recognize the importance of engaging in open and collaborative research efforts to address these challenges. By fostering dialogue and knowledge-sharing among researchers, we can accelerate progress towards safe and beneficial AI while ensuring that the development of these technologies is guided by diverse perspectives and expertise. This may involve the creation of shared research platforms, open-source tools, and standardized benchmarks to facilitate reproducibility and comparative analysis of different approaches. By addressing these limitations and focusing on responsible development, we hope to pave the way for beneficial contributions to the field that can leverage the full potential of AI while safeguarding against its risks. Through a combination of theoretical modeling, empirical experimentation, and multidisciplinary collaboration, we aim to create a foundation for the safe and productive exploration of self-improving AI systems, ultimately contributing to the realization of the transformative potential of artificial intelligence in a manner that benefits humanity as a whole.

References

[A+23a] J. Achiam et al. Gpt-4 technical report. Technical report, OpenAI, 2023.

[A+23b] R. Anil et al. Gemini: A family of highly capable multimodal models. Technical report, Google, 2023.

[B+20] T. Brown et al. Language models are few-shot learners. *NeurIPS 2020*, 2020.

[BBL+23] B. Bischl, M. Binder, M. Lang, T. Pielok, J. Richter, S. Coors, J. Thomas, T. Ullmann, M. Becker, A.L. Boulesteix, D. Deng, and M. Lindauer. Hyperparameter optimization:

Foundations, algorithms, best practices, and open challenges. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2023.

[Bie23] C. Biever. Chatgpt broke the turing test – the race is on for new ways to assess ai. *Nature*, 2023.

[BLH21] Y. Bengio, Y. Lecun, and G. Hinton. Deep learning for ai. *Communications of the ACM*, 2021.

[Bos07] N. Bostrom. Technological revolutions: Ethics and policy in the dark. *Nanoscale: Issues and Perspectives for the Nano Century*, 2007.

[Bos12] N. Bostrom. The superintelligent will: Motivation and instrumental rationality in advanced artificial agents. *Minds and Machines*, 2012.

[CB12] C.Shulman and N. Bostrom. How hard is artificial intelligence? evolutionary arguments and selection effects. *Journal of Consciousness Studies*, 2012.

[Cha10] D.J. Chalmers. The singularity: A philosophical analysis. *Journal of Consciousness Studies*, 2010.

[CLSZ23] X. Chen, M. Lin, N. Schärli, and D. Zhou. Teaching large language models to self-debug. *arXiv:2304.05128 [cs.CL]*, 2023.

[CLTB+17] P.F. Christiano, J. Leike, M. Martic T. Brown, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. *Advances in Neural Information Processing Systems*, 2017.

[DCLT19] J. Devlin, M. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019.

[DLD+22] Q. Dong, L. Li, D. Dai, C. Zheng, Z. Wu, B. Chang, X. Sun, J. Xu, L. Li, and Z. Sui. A survey on in-context learning. *arXiv:2301.00234 [cs.CL]*, 2022.

[DWH+22] M. Deng, J. Wang, C.P. Hsieh, Y. Wang, H. Guo, T. Shu, M. Song, E. Xing, and Z. Hu. Rlprompt: Optimizing discrete text prompts with reinforcement learning. *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

[EMH19] T. Elsken, J.H. Metzen, and F. Hutter. Neural architecture search: A survey. *Journal of Machine Learning Research*, 2019.

[EMS+20] N. Erickson, J. Mueller, A. Shirkov, H. Zhang, P. Larroy, M. Li, and A. Smola. Autogluon-tabular: Robust and accurate automl for structured data. *arXiv:2003.06505 [stat.ML]*, 2020.

[FBM+23] C. Fernando, D. Banarse, H. Michalewski, S. Osindero, and T. Rocktäschel. Promptbreeder: Self-referential self-improvement via prompt evolution. *arXiv:2309.16797 [cs.CL]*, 2023.

[FEF+22] M. Feurer, K. Eggensperger, S. Falkner, M. Lindauer, and F. Hutter. Auto-sklearn 2.0: Hands-free automl via meta-learning. *Journal of Machine Learning Research 23 (2022)*, 2022.

[FH19] M. Feurer and F. Hutter. Hyperparameter optimization. *Automated Machine Learning*, 2019.

[FKE+15] M. Feurer, A. Klein, K. Eggensperger, J. Springenberg, M. Blum, and F. Hutter. Efficient and robust automated machine learning. *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, 2015.

[FKE+19] M. Feurer, A. Klein, K. Eggensperger, J.T. Springenberg, M. Blum, and F. Hutter. Auto-sklearn: Efficient and robust automated machine learning. *Automated Machine Learning*, 2019.

[GCK16] H. Guo, H.K. Cheng, and K. Kelley. Impact of network structure on malware propagation: A growth curve perspective. *Journal of Management Information Systems 33*, 2016.

[Goe14] B. Goertzel. Artificial general intelligence: Concept, state of the art, and future prospects. *Journal of Artificial General Intelligence*, 2014.

[Goo59] I.J. Good. Speculations on perceptrons and other automata. *RC115*, 1959.

[Goo65] I.J. Good. Speculations concerning the first ultraintelligent machine. *Advances in Computers*, 1965.

[Goo70] I.J. Good. Some future social repercussions of computers. *International Journal of Environmental Studies*, 1970.

[HGH+23] J. Huang, S. Gu, L. Hou, Y. Wu, X. Wang, H. Yu, and J. Han. Large language models can self-improve. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[HKV19] F. Hutter, L. Kotthoff, and J. Vanschoren. Automated machine learning: Methods, systems, challenges. *Springer*, 2019.

[HSLS23] O. Honovich, T. Scialom, O. Levy, and T. Schick. Unnatural instructions: Tuning language models with (almost) no human labor. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023.

[Hut10] M. Hutter. Can intelligence explode? *Journal of Consciousness Studies*, 2010.

[KGR+22] T. Kojima, S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa. Large language models are zero-shot reasoners. *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, 2022.

[KTH+17] L. Kotthoff, C. Thornton, H.H. Hoos, F. Hutter, and K. Leyton-Brown. Auto-weka 2.0: Automatic model selection and hyperparameter optimization in weka. *Journal of Machine Learning Research*, 2017.

[KTH+19] L. Kotthoff, C. Thornton, H.H. Hoos, F. Hutter, and K. Leyton-Brown. Auto-weka: Automatic model selection and hyperparameter optimization in weka. *Automated Machine Learning*, 2019.

[Kur05] R. Kurzweil. The singularity is near: When humans transcend biology. *Viking Press*, 2005.

[LH07] S. Legg and M. Hutter. A collection of definitions of intelligence. *Advances in Artificial General Intelligence: Concepts, Architectures and Algorithms—Proceedings of the AGI Workshop 2006*, 2007.

[LYZ+23] X. Li, P. Yu, C. Zhou, T. Schick, O. Levy, L. Zettlemoyer, J. Weston, and M. Lewis. Self-alignment with instruction backtranslation. *arXiv:2308.06259 [cs.CL]*, 2023.

[LZD+23] X. Li, T. Zhang, Y. Dubois, R. Taori, I. Gulrajani, C. Guestrin, P. Liang, and T.B. Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models. *GitHub repository* https://github.com/tatsu-lab/alpaca_eval, 2023.

[Mal98] T. Malthus. An essay on the principle of population. 1798.

[Min66] M. Minsky. Artificial intelligence. *Scientific American*, 1966.

[MS12] L. Muehlhauser and A. Salamon. Intelligence explosion: Evidence and import. *Singularity Hypotheses: A Scientific and Philosophical Assessment.*, 2012.

[MSdF+23] M.R. Morris, J. Sohl-dickstein, N. Fiedel, T. Warkentin, A. Dafoe, A. Faust, C. Farabet, and S. Legg. Levels of agi: Operationalizing progress on the path to agi. *arXiv:2311.02462 [cs.AI]*, 2023.

[Neu66] J. Neumann. Theory of self-reproducing automata. *University of Illinois Press*, 1966.

[Nil09] N.J. Nilsson. The quest for artificial intelligence. *Cambridge University Press*, 2009.

[Omo07] S. Omohundro. The nature of self-improving artificial intelligence. *Singularity Summit*, 2007.

[Omo08] S. Omohundro. The basic ai drives. *Proceedings of the 2008 conference on Artificial General Intelligence 2008: Proceedings of the First AGI Conference*, 2008.

[Omo13] S. Omohundro. Rational artificial intelligence for the greater good. *Singularity Hypotheses*, 2013.

[PHZB23] A. Prasad, P. Hase, X. Zhou, and M. Bansal. Grips: Gradient-free, edit-based instruction search for prompting large language models. *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, 2023.

[R+23] B. Rozière et al. Code llama: Open foundation models for code. *arXiv:2308.12950 [cs.CL]*, 2023.

[RHW86] D.E. Rumelhart, G.E. Hinton, and R.J. Williams. The perceptron: A probabilistic model for information storage and organization in the brain. *Nature*, 1986.

[Ros58] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 1958.

[RSM+23] R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C.D. Manning, and C. Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv:2305.18290 [cs.LG]*, 2023.

[RWC+19] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019.

[SB10] A. Sandberg and N. Bostrom. Whole brain emulation: A roadmap. Technical report, Future of Humanity Institute, University of Oxford, 2010.

[Sch07] J. Schmidhuber. Gödel machines: Fully self-referential optimal universal self-improvers. *Artificial General Intelligence.*, 2007.

[SMZ+23] A. Shypula, A. Madaan, Y. Zeng, U. Alon, J. Gardner, M. Hashemi, G. Neubig, P. Ranganathan, O. Bastani, and A. Yazdanbakhsh. Learning performance-improving code edits. *arXiv:2302.07867 [cs.SE]*, 2023.

[SOW+20] N. Stiennon, L. Ouyang, J. Wu, D. Ziegler, R. Lowe, C. Voss, A. Radford, D. Amodei, and P.F. Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, 2020.

[SRI+20] T. Shin, Y. Razeghi, R. L. Logan IV, E. Wallace, and S. Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

[SSZ+23] Z. Sun, Y. Shen, Q. Zhou, H. Zhang, Z. Chen, D. Cox, Y. Yang, and C. Gan. Principle-driven self-alignment of language models from scratch with minimal human supervision. *arXiv:2305.03047 [cs.LG]*, 2023.

[ST99] I. Seidl and C.A. Tisdell. Carrying capacity reconsidered: from malthus' population theory to cultural carrying capacity. *Ecological Economics*, 1999.

[SYD+23] T. Schick, J.D. Yu, R. Dessì, R. Raileanu, M. Lomeli, L. Zettlemoyer, N. Cancedda, and T. Scialom. Toolformer: Language models can teach themselves to use tools. *arXiv:2302.04761 [cs.CL]*, 2023.

[SZ03] G. Serazzi and S. Zanero. Computer virus propagation models. *International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS 2003)*, 2003.

[T+23] H. Touvron et al. Llama: Open and efficient foundation language models. Technical report, Meta, 2023.

[TDE+23] A. Tornede, D. Deng, T. Eimer, J. Giovanelli, A. Mohan, T. Ruhkopf, S. Segel, D. Theodorakopoulos, T. Tornede, H. Wachsmuth, and M. Lindauer. Automl in the age of large language models: Current challenges, future opportunities and risks. *arXiv:2306.08107 [cs.LG]*, 2023.

[THHLB12] C. Thornton, F. Hutter, H.H. Hoos, and K. Leyton-Brown. Auto-weka: Combined selection and hyperparameter optimization of classification algorithms. *arXiv:1208.3719 [cs.LG]*, 2012.

[Tur50] A. Turing. Computing machinery and intelligence. *Mind*, 1950.

[Tur51] A. Turing. Intelligent machinery, a heretical theory. *A lecture given to '51 Society' at Manchester*, 1951.

[Ver38] P.F. Verhulst. Notice sur la loi que la population suit dans son accroissement. correspondance mathematique et physique publiee par a. *Quetelet*, 1838.

[Vin93] V. Vinge. The coming technological singularity: How to survive in the post-human era. *21: Interdisciplinary Science and Engineering in the Era of Cyberspace*, 1993.

[VSP+17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems.*, 2017.

[WKM+23] Y. Wang, Y. Kordi, S. Mishra, A. Liu, N.A. Smith, D. Khashabi, and H. Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023.

[WRP19] M. Wistuba, A. Rawat, and T. Pedapati. A survey on neural architecture search. *arXiv:1905.01392 [cs.LG]*, 2019.

[WSS+23] C. White, M. Safari, R. Sukthanker, B. Ru, T. Elsken, A. Zela, D. Dey, and F. Hutter. Neural architecture search: Insights from 1000 papers. *arXiv:2301.08727 [cs.LG]*, 2023.

[WWS+22] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q.V. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, 2022.

[Yam15] R.V. Yampolskiy. From seed ai to technological singularity via recursively self-improving software. *arXiv:1502.06512 [cs.AI]*, 2015.

[YDY+19] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R.R. Salakhutdinov, and Q.V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in Neural Information Processing Systems 32*, 2019.

[YPC+24] W. Yuan, R.Y. Pang, K. Cho, X. Li, S. Sukhbaatar, J. Xu, and J. Weston. Self-rewarding language models. *arXiv:2401.10020 [cs.CL]*, 2024.

[Yud08a] E. Yudkowsky. Artificial intelligence as a positive and negative factor in global risk. *Global Catastrophic Risks*, 2008.

[Yud08b] E. Yudkowsky. Efficient cross-domain optimization. *http://lesswrong.com/lw/vb/efficient_crossdomain_optimization*, 2008.

[YWL+23] C. Yang, X. Wang, Y. Lu, H. Liu, Q. V. Le, D. Zhou, and X. Chen. Large language models as optimizers. *arXiv:2309.03409 [cs.LG]*, 2023.

[YYH23] H. Yang, S. Yue, and Y. He. Auto-gpt for online decision making: Benchmarks and additional opinions. *arXiv:2306.02224 [cs.AI]*, 2023.

[ZLH21] L. Zimmer, M. Lindauer, and F. Hutter. Auto-pytorch: Multi-fidelity metalearning for efficient and robust autodl. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[ZLMK23] E. Zelikman, E. Lorch, L. Mackey, and A.T. Kalai. Self-taught optimizer (stop): Recursively self-improving code generation. *arXiv:2310.02304 [cs.CL]*, 2023.

[ZMH+23] Y. Zhou, A. I. Muresanu, Z. Han, K. Paster, S. Pitis, H. Chan, and J. Ba. Large language models are human-level prompt engineers. *In InternationalConference on Learning Representations (ICLR)*, 2023.

[ZWMG22] E. Zelikman, Y. Wu, J. Mu, and N.D. Goodman. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, 2022.

Appendix A The basic RPF and RSI prompt

```
Review the user's question and the corresponding response using the additive 5-point
scoring system described below. Points are accumulated based on the satisfaction of each
criterion:
- Add 1 point if the response is relevant and provides some information related to
the user's inquiry, even if it is incomplete or contains some irrelevant content.
- Add another point if the response addresses a substantial portion of the user's question,
but does not completely resolve the query or provide a direct answer.
- Award a third point if the response answers the basic elements of the user's question in a
useful way, regardless of whether it seems to have been written by an AI Assistant or if it
has elements typically found in blogs or search results.
- Grant a fourth point if the response is clearly written from an AI Assistant's perspective,
addressing the user's question directly and comprehensively, and is well-organized and
helpful, even if there is slight room for improvement in clarity, conciseness or focus.
- Bestow a fifth point for a response that is impeccably tailored to the user's question
by an AI Assistant, without extraneous information, reflecting expert knowledge, and
demonstrating a high-quality, engaging, and insightful answer.
User: <INSTRUCTION_HERE>
<response><RESPONSE_HERE></response>
After examining the user's instruction and the response:
- Briefly justify your total score, up to 100 words.
- Conclude with the score using the format: \Score: <total points>"
Remember to assess from the AI Assistant perspective, utilizing web search knowledge as
necessary. To evaluate the response in alignment with this additive scoring model, we'll
systematically attribute points based on the outlined criteria.
```

RPF prompt from [YPC+24].

```
from helpers import extract_code

def improve_algorithm(initial_solution, utility,
,→ language_model):
"""Improves a solution according to a utility
```

```
,→ function."""
expertise = "You are an expert computer science
,→ researcher and programmer, especially skilled
,→ at optimizing algorithms."
message = f"""Improve the following solution:
```python
{initial_solution}
```

You will be evaluated based on this score function:
```python
{utility.str}
```

You must return an improved solution. Be as creative as
,→ you can under the constraints.
Your primary improvement must be novel and non-trivial.
,→ First, propose an idea, then implement it."""
n_messages = min(language_model.
,→ max_responses_per_call, utility.budget)
new_solutions = language_model.batch_prompt(
,→ expertise, [message] * n_messages, temperature
,→ =0.7)
new_solutions = extract_code(new_solutions)
best_solution = max(new_solutions, key=utility)
return best_solution
```

RSI prompt from [ZLMK23].