Reinforcement Learning for Temporal Reasoning and Sequential Decision-Making in Language Models

Ermenegildo Farinelli
Department of Computer Science, University of North Texas, Denton, TX, USA.
ermenegildo1971@unt.edu

Susannah Harkins
Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO, USA.
susannah909@missouri.edu

Manan Naidu
Department of Computer Science, University of Houston, Houston, TX, USA.
manan.work@uh.edu

Abstract

Reinforcement learning has become a central mechanism for aligning large language models with complex objectives, yet its role in temporal reasoning and sequential decision-making remains under-theorized from a systems perspective. This paper examines how reinforcement learning architectures, reward modeling pipelines, inference-time control, and deployment infrastructures interact when language models are asked to reason across time, plan over multiple steps, and act within partially observable environments. It argues that temporal reasoning in language models is not merely a prompt-engineering problem but a socio-technical capability shaped by credit assignment, trajectory filtering, memory design, latency budgets, governance structures, and evaluation regimes. The discussion compares policy-gradient methods, preference optimization, verbal reinforcement, and tool-augmented agent loops, highlighting trade-offs among sample efficiency, interpretability, robustness, and computational sustainability. Special attention is given to small-parameter models, where scalable reasoning path filtering and alignment can reduce training costs but may also constrain representational capacity. The paper further considers fairness, safety, and policy implications when sequential language agents mediate decisions in education, healthcare, finance, and public administration. It concludes that responsible temporal reasoning requires layered governance, reproducible evaluation, energy-aware infrastructure, and institutions capable of auditing multi-step behavior rather than isolated outputs.

Keywords

reinforcement learning; temporal reasoning; large language models; sequential decision-making; alignment; AI governance; sustainable AI; robustness; fairness.

1. Introduction

Language models have moved from static text predictors toward interactive systems that plan, call tools, revise their own outputs, and participate in multi-step workflows. This transition has made temporal reasoning and sequential decision-making central concerns for both research and deployment. A model that can summarize a document in one response is not necessarily capable of maintaining a coherent plan across hours of interaction, adapting to

delayed feedback, or recognizing when earlier assumptions have become invalid. Reinforcement learning offers a framework for shaping such behavior because it explicitly addresses reward-driven adaptation over time rather than only next-token prediction [1]. However, applying reinforcement learning to language models introduces distinctive problems of credit assignment, instability, reward misspecification, and infrastructure cost that cannot be resolved by algorithmic novelty alone [3]. The resulting systems are socio-technical arrangements in which model weights, reward models, human annotators, evaluation harnesses, serving stacks, and organizational policies jointly determine what counts as good sequential behavior.

The intellectual foundations of this area combine sequence modeling, decision theory, and preference learning. Transformer architectures provide the representational substrate for long-range dependencies, but they do not by themselves guarantee temporal coherence or strategic consistency [2]. Reinforcement learning from human feedback demonstrated that preference data can be used to train reward models and optimize language model policies toward helpful and harmless behavior [4]. Subsequent work has explored direct preference optimization, constitutional approaches, and scalable oversight, each with different assumptions about how values are elicited, represented, and enforced [5]. These methods are often presented as alignment techniques, but from a systems perspective they are also control mechanisms that determine how language models allocate attention, when they stop reasoning, and how they recover from error.

Temporal reasoning in language models is particularly challenging because the model must often operate in partially observable environments where the full state is not available in a single context window. The model may need to infer hidden variables, track commitments, respect deadlines, and coordinate with external agents. Sequential decision-making therefore involves not only choosing the next token but also deciding when to search, when to ask for clarification, when to invoke an external tool, and when to defer to a human. These decisions have latency, financial, and safety consequences. A system that reasons carefully but slowly may be unusable in real-time settings, while a system that acts quickly may accumulate errors that are difficult to detect. Reinforcement learning can optimize such trade-offs, but only if the reward signal captures long-horizon consequences rather than immediate surface features [6].

This paper adopts a system-level orientation. It does not propose a new algorithm or derive formal guarantees. Instead, it analyzes structural trade-offs among architecture, governance, infrastructure, deployment, sustainability, robustness, fairness, and policy. The argument is that temporal reasoning capabilities in language models emerge from layered interactions: policy optimization shapes behavior, reward models encode values, memory systems preserve state, serving infrastructure constrains latency, and institutions determine accountability. Ignoring any layer produces brittle systems that may perform well in demonstrations but fail in production. The discussion therefore moves from foundational concepts to sequential reasoning, architecture, tool use, alignment, fairness, infrastructure, and policy implications.

2. Foundations of Reinforcement Learning and Language Modeling

Reinforcement learning is conventionally understood as learning from interaction, where an agent receives observations, takes actions, and receives rewards that may be delayed or sparse. Classic treatments emphasize exploration, exploitation, value estimation, and policy improvement [1]. Deep reinforcement learning extended these ideas to high-dimensional sensory inputs, first in games and control tasks [7], and later in complex strategic domains

such as Go [6]. The success of these systems depended on large-scale computation, careful reward design, and the ability to simulate experience. Language models differ from these domains because their action space is discrete and enormous, their environments are often linguistic rather than physical, and their rewards are frequently learned from human preferences rather than hand-coded.

Language modeling itself can be framed as sequential prediction, where the model estimates the probability of the next token given previous tokens. Transformer architectures made this prediction task highly scalable by replacing recurrence with attention mechanisms that capture long-range dependencies [2]. Scaling laws and few-shot learning further showed that large models could perform many tasks without task-specific parameter updates [3]. Yet next-token prediction alone does not provide a mechanism for optimizing long-term outcomes. A model may generate plausible text while failing to follow a plan, respect constraints, or update beliefs when new information arrives. Reinforcement learning supplies a complementary optimization loop in which generated sequences are evaluated and used to adjust the policy.

Preference-based reinforcement learning has been especially influential. In this approach, human comparisons between candidate outputs are used to train a reward model, which then guides policy optimization [5]. This framework underlies reinforcement learning from human feedback and related methods for instruction following [4]. Proximal policy optimization became a common optimizer because it balances stability and sample efficiency [8]. More recent work has explored direct preference optimization, which bypasses explicit reward modeling by optimizing the policy directly on preference pairs [9]. Each approach embodies different trade-offs: reward modeling offers interpretability and reuse but introduces a learned proxy that can be gamed; direct optimization simplifies the pipeline but may reduce transparency about what values are being encoded.

For temporal reasoning, the key issue is credit assignment across many steps. A language model may produce a long chain of reasoning, invoke several tools, and generate a final answer. If the final answer is wrong, which intermediate decision should be penalized? If the final answer is right, did the model use a valid method or merely guess? Reinforcement learning methods that treat the entire sequence as one action struggle with this problem. Methods that decompose reasoning into smaller steps can provide denser feedback, but they require reliable intermediate rewards or human oversight at a granularity that is often impractical. This tension between sparse outcome rewards and dense process rewards is one of the central structural challenges in applying reinforcement learning to temporal reasoning in language models [14].

3. Temporal Reasoning as a Sequential Decision Problem

Temporal reasoning requires a model to represent change, duration, ordering, causality, and uncertainty. In natural language, temporal information is often implicit, ambiguous, or distributed across multiple turns. A user may say that a task is urgent without specifying a deadline, or refer to an earlier decision without restating its conditions. A language model engaged in sequential decision-making must maintain an internal or external state that tracks these commitments. When that state is held only in the context window, it is vulnerable to truncation and interference. When it is stored externally, the model must decide what to write, when to retrieve it, and how to reconcile conflicting memories. These are decision problems, not merely memory problems.

Chain-of-thought prompting demonstrated that encouraging models to produce intermediate reasoning steps can improve performance on tasks that require multi-step inference [13]. However, prompting alone does not guarantee that the generated reasoning is faithful or that the model will use it consistently. ReAct showed that interleaving reasoning with actions can improve performance in interactive tasks by allowing the model to gather information and adjust its plan [11]. Reflexion introduced a form of verbal reinforcement in which the model reflects on failures and uses those reflections to improve subsequent attempts [12]. These methods illustrate a spectrum between internal deliberation and external interaction. They also reveal that temporal reasoning is closely tied to control: the model must decide when to think, when to act, and when to revise.

From a reinforcement learning perspective, temporal reasoning can be modeled as a partially observable sequential decision process. The agent does not directly observe the full state of the world, and its actions affect future observations. Rewards may be delayed, and the environment may include other agents whose behavior is uncertain. This framing highlights the importance of exploration. A language model may need to try different reasoning paths, query different tools, or ask clarifying questions to reduce uncertainty. However, exploration in language models is expensive and potentially risky. Generating many candidate trajectories increases inference cost and may expose users to inappropriate content. The system designer must therefore choose between broad exploration during training, constrained exploration during inference, and post-hoc filtering of trajectories.

Temporal abstraction is another structural concern. Human reasoning often operates at multiple time scales: immediate actions, short-term plans, and long-term goals. A language model that reasons only at the token level may fail to represent these levels explicitly. Hierarchical reinforcement learning offers concepts for temporal abstraction, but applying them to language models requires mechanisms for proposing subgoals, evaluating progress, and switching strategies. In practice, many agent frameworks implement a form of hierarchy through prompt scaffolding, where a planner generates steps and an executor carries them out. This division of labor can improve reliability, but it also creates new failure modes when the planner and executor disagree or when the environment changes between planning and execution. The governance challenge is to make these transitions auditable and correctable.

4. Architectures for Reinforcement Learning with Language Models

Architectures for reinforcement learning with language models typically combine a policy model, a reward or preference model, a training loop, and an inference-time controller. The policy model generates text or actions. The reward model assigns scores to outputs or trajectories. The training loop updates the policy using policy gradients, preference optimization, or related methods. The inference-time controller decides how many samples to draw, how to filter them, and when to stop. Each component introduces design choices that affect temporal reasoning. For example, a reward model that evaluates only final answers may encourage shortcut reasoning, while a reward model that evaluates intermediate steps may require costly annotation and may impose a particular reasoning style.

Proximal policy optimization has been widely used because it constrains policy updates and improves stability in large-scale training [8]. Direct preference optimization simplifies the pipeline by training directly on preference data, but it may offer less explicit control over the reward signal [9]. Reinforcement learning from human feedback with reward modeling remains attractive because it separates value learning from policy optimization, allowing organizations to inspect and revise reward models [4]. However, learned reward models can

be exploited by policies that discover unintended patterns. This problem, often discussed as reward hacking or specification gaming, is especially dangerous in temporal settings because a model may learn to satisfy a proxy metric at the expense of long-term user interests [16].

Recent work on scalable reasoning path filtering and alignment for smaller parameter models illustrates how architecture and compute constraints interact. Filtering candidate reasoning paths before policy optimization can reduce the cost of reinforcement learning and make alignment more feasible for models that cannot afford massive sampling budgets [15]. The approach is attractive for deployment in edge environments, privacy-sensitive settings, and organizations with limited computational resources. At the same time, aggressive filtering may discard unusual but valid reasoning strategies, reducing diversity and potentially harming robustness. Small models also have less capacity to represent complex temporal dependencies, so gains from filtering may be uneven across tasks. The architectural trade-off is therefore not simply efficiency versus accuracy; it also involves coverage, fairness, and the risk of reinforcing dominant reasoning styles.

Inference-time architectures are equally important. A model may be trained with reinforcement learning but deployed with search, self-consistency, or tool-use policies that were not part of training. If the training reward assumed a single generation but deployment samples many candidates, the effective behavior can shift. Similarly, if the model is trained to reason for a fixed number of steps but deployment imposes a strict latency budget, the model may truncate reasoning in ways that were never optimized. These mismatches are a form of technical debt in machine learning systems [20]. They suggest that reinforcement learning for temporal reasoning should be co-designed with serving infrastructure, monitoring, and rollback mechanisms rather than treated as an offline training problem.

5. Sequential Decision-Making in Tool Use and Embodied Interaction

Tool use is one of the most concrete settings for sequential decision-making in language models. A model may call a search engine, query a database, execute code, or interact with an application programming interface. Each action changes the information available and may have side effects. The model must decide which tool to use, how to formulate the query, how to interpret the result, and whether to continue or stop. ReAct provides a general pattern for interleaving reasoning and action, but it does not solve the underlying control problem [11]. In production, tool use requires permissions, rate limits, error handling, and audit trails. A model that can execute code or modify records introduces security and accountability risks that are qualitatively different from text generation.

Embodied interaction extends these challenges to physical or simulated environments. A language model may guide a robot, control a simulation, or coordinate with sensors. Here, temporal reasoning must contend with real-time constraints, noisy observations, and safety-critical consequences. Reinforcement learning has a long history in control and robotics, but language models add a layer of semantic abstraction that can be both powerful and unpredictable [7]. A model might interpret a vague instruction in a way that is locally plausible but globally unsafe. It might also fail to recognize that a physical action cannot be undone. The system architecture must therefore include safety envelopes, human override, and conservative action selection. These are governance mechanisms as much as technical ones.

Multi-agent settings further complicate sequential decision-making. Language models may interact with other language models, human users, or conventional software agents. The

environment becomes strategic, because each agent's behavior depends on expectations about others. Cooperative AI research highlights problems of coordination, trust, and norm emergence that cannot be reduced to individual reward maximization [24]. A language model trained to maximize a narrow reward may exploit other agents or create negative externalities. Temporal reasoning in such settings requires representations of commitments, reputations, and joint plans. It also requires institutions that can enforce agreements and resolve conflicts. Without these, sequential decision-making may scale in capability while degrading in social reliability.

6. Alignment, Reward Modeling, and Governance

Alignment is often described as the problem of ensuring that artificial intelligence systems act in accordance with human values and intentions. In reinforcement learning with language models, alignment is operationalized through reward models, preference data, constitutions, and oversight processes [4]. These mechanisms are not neutral. They encode particular conceptions of helpfulness, harmlessness, honesty, and propriety. They also reflect the distribution of annotators, the incentives of organizations, and the legal environments in which systems are deployed. A reward model trained on one population may misrepresent the values of another, leading to temporal decisions that are unfair or culturally inappropriate [21]. Governance therefore begins at the data and reward design stage, not only at deployment.

Scalable oversight is a central governance challenge for temporal reasoning. If a model can perform tasks that are difficult for humans to evaluate, then human feedback may be insufficient to detect subtle errors. Reward modeling research has explored ways to use human judgments efficiently, but it cannot eliminate the need for institutions that can investigate failures [14]. In multi-step settings, oversight must consider not only final outputs but also the trajectory: what information was gathered, what alternatives were considered, and what side effects occurred. This requires logging, provenance, and interpretability tools that are often absent from current language model deployments. It also requires legal and organizational frameworks that assign responsibility when an autonomous sequence of actions causes harm.

The policy implications are significant. Regulators may need to require documentation of reward models, evaluation of sequential behavior, and mechanisms for contesting automated decisions. Algorithmic ethics scholarship has long emphasized transparency, accountability, and due process [21]. These principles become more complex when decisions emerge from temporal interaction rather than a single classification. A language model may make a series of seemingly reasonable choices that collectively produce an discriminatory outcome. Fairness audits that examine only individual outputs may miss this pattern. Temporal fairness therefore demands new evaluation methods that track outcomes across time and across groups. It also demands governance structures capable of updating rules as environments change.

7. Robustness, Fairness, and Evaluation

Robustness in reinforcement learning is difficult even in controlled environments. Deep reinforcement learning has been shown to be sensitive to hyperparameters, random seeds, and implementation details, making reproducibility a persistent concern [23]. Language models add further sources of variance: prompt phrasing, sampling temperature, context length, and external tool availability. A temporal reasoning system may appear robust in a benchmark but fail when the environment changes slightly. For example, a model trained to schedule appointments may fail when a user changes time zones or when a calendar system returns an

unexpected format. Robustness requires not only better training but also runtime monitoring, fallback behavior, and the ability to detect distribution shift.

Fairness is similarly multidimensional. Foundation models can perpetuate and amplify biases present in training data, and their scale makes auditing difficult [19]. In sequential decision-making, bias can compound over time. A model that gives slightly less accurate information to one group may cause that group to make worse decisions, which in turn produces feedback that reinforces the model's initial bias. Temporal fairness therefore cannot be reduced to parity in a single response. It requires tracking how benefits and harms accumulate across interactions. It also requires attention to representation and voice: whose preferences are included in reward models, whose histories are present in training data, and whose errors are treated as acceptable [18].

Evaluation is the connective tissue between technical performance and governance. Current benchmarks often emphasize final-answer accuracy, but temporal reasoning requires process evaluation. Did the model maintain a coherent plan? Did it update beliefs when new evidence arrived? Did it respect constraints? Did it seek help when appropriate? These questions are difficult to operationalize, and they require evaluation regimes that combine automated metrics, human review, and adversarial testing. Reproducibility is essential because claims about robustness and fairness must be verifiable [23]. Evaluation must also be ongoing, because deployed systems encounter new situations that were not represented in static test sets. Without continuous evaluation, temporal reasoning systems may drift into unsafe or unfair behavior long after initial approval.

8. Infrastructure, Deployment, and Sustainability

The infrastructure required for reinforcement learning with language models is substantial. Training reward models, sampling trajectories, and optimizing policies consume large amounts of computation and energy. The environmental impact of deep learning has become a subject of policy concern, with attention to carbon emissions, water use, and supply chain dependencies [22]. Temporal reasoning systems may be especially demanding because they generate multiple reasoning paths, call external tools, and maintain long contexts. Efficiency is therefore not only an economic concern but also an ethical and environmental one. Designers must weigh the benefits of additional reasoning against the costs of computation and latency.

Deployment architecture shapes what kinds of temporal reasoning are feasible. Cloud-based systems can scale resources but may raise privacy and sovereignty issues. Edge-based systems can reduce latency and protect data but have limited memory and compute. Small-parameter models are attractive in edge and privacy-sensitive settings, and methods that filter reasoning paths can improve their alignment efficiency [15]. However, small models may struggle with complex temporal abstraction. A hybrid architecture may be necessary, where a small local model handles routine decisions and a larger remote model is consulted for difficult cases. Such architectures introduce new coordination problems: how to hand off state, how to preserve privacy, and how to ensure consistent policy across components.

Sustainability also includes organizational sustainability. Machine learning systems accumulate technical debt through entangled dependencies, undocumented data pipelines, and hidden feedback loops [20]. A temporal reasoning system that relies on external tools, proprietary reward models, and rapidly changing application programming interfaces is fragile. It may cease to function when a tool changes, a vendor disappears, or a policy shifts.

Sustainable deployment requires modular design, clear interfaces, version control, and contingency planning. It also requires investment in monitoring and maintenance, which are often underfunded relative to initial development. The long-term viability of reinforcement learning for language models depends on treating infrastructure as a first-class research and governance concern.

9. Policy Implications and Institutional Design

Policy frameworks for artificial intelligence must address systems that act over time, not only systems that produce single outputs. A language model that schedules, negotiates, or manages resources may affect individuals and communities in ways that are difficult to trace. Regulatory approaches based on transparency, accountability, and contestability can provide a foundation, but they need adaptation for temporal systems [21]. For example, a right to explanation may require a narrative account of a sequence of decisions, not merely a description of the final model. A right to human review may require the ability to pause or reverse an ongoing process. These are institutional design problems as much as technical ones.

Institutions also shape the incentives for responsible development. If organizations are rewarded only for capability benchmarks, they may underinvest in robustness, fairness, and sustainability. If liability rules are unclear, they may deploy systems without adequate oversight. If procurement processes favor the lowest cost, they may select systems that externalize harms onto users and workers. Policy can correct these incentives through standards, audits, insurance requirements, and public procurement criteria. It can also support public research infrastructure that reduces dependence on a few large firms. The governance of temporal reasoning should therefore be pluralistic, involving researchers, affected communities, regulators, and industry.

Cross-domain comparisons are instructive. In healthcare, sequential language agents might assist with triage or care coordination, where errors can be life-threatening. In finance, they might execute trades or assess credit, where fairness and market stability are at stake. In education, they might tutor students over months, where long-term developmental effects matter. In public administration, they might process benefits or permits, where due process and non-discrimination are legal requirements. Each domain has existing governance structures, professional norms, and accountability mechanisms. Effective deployment requires integrating with these structures rather than treating them as obstacles. The same reinforcement learning method may be appropriate in one domain and unacceptable in another, depending on the severity of harm and the availability of human oversight.

10. Conclusion

Reinforcement learning offers a powerful framework for temporal reasoning and sequential decision-making in language models, but its benefits depend on system-level design. The technical core involves policy optimization, reward modeling, preference learning, and trajectory filtering. The surrounding layers involve memory, tool use, serving infrastructure, evaluation, governance, and policy. Failures can arise at any layer, and they often compound over time. A model that is well aligned in a single turn may become unreliable over a long interaction. A system that is efficient in the lab may be unsustainable in production. A reward model that reflects one population may disadvantage another. These are not isolated issues; they are structural features of deploying adaptive language agents in open-ended environments.

The paper has argued for a layered approach that treats temporal reasoning as a socio-technical capability. Researchers should develop methods that improve credit assignment, support temporal abstraction, and evaluate process as well as outcomes. Engineers should co-design training and inference so that latency budgets, tool availability, and safety constraints are represented in the optimization loop. Organizations should invest in monitoring, logging, and maintenance rather than treating deployment as the end of development. Policymakers should create frameworks that can audit multi-step behavior, assign accountability, and protect affected communities. Together, these efforts can help ensure that reinforcement learning for language models enhances human decision-making without creating new forms of opacity, inequity, or environmental harm.

References

1. Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction (2nd ed.). MIT Press.

2. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30, 5998–6008.

3. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... Amodei, D. (2020). Language models are few-shot learners. Advances in Neural Information Processing Systems, 33, 1877–1901.

4. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems, 35, 27730–27744.

5. Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., & Amodei, D. (2017). Deep reinforcement learning from human preferences. Advances in Neural Information Processing Systems, 30, 4299–4307.

6. Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., ... Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. Nature, 529(7587), 484–489.

7. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... Hassabis, D. (2015). Human-level control through deep reinforcement learning. Nature, 518(7540), 529–533.

8. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.

9. Rafailov, R., Sharma, A., Mitchell, E., Ermon, S., Manning, C. D., & Finn, C. (2023). Direct preference optimization: Your language model is secretly a reward model. Advances in Neural Information Processing Systems, 36, 53728–53741.

10. Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., ... Kaplan, J. (2022). Training a helpful and harmless assistant with reinforcement learning from human feedback. arXiv preprint arXiv:2204.05862.

11. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2023). ReAct: Synergizing reasoning and acting in language models. International Conference on Learning Representations.

12. Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., & Yao, S. (2023). Reflexion: Language agents with verbal reinforcement learning. Advances in Neural Information Processing Systems, 36, 8634–8652.

13. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., ... Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824–24837.

14. Leike, J., Krueger, D., Everitt, T., Martic, M., Maini, V., & Legg, S. (2018). Scalable agent alignment via reward modeling: A research direction. arXiv preprint arXiv:1811.07871.

15. Li, Q. (2026, May). TrajLite PPO: Scalable Reasoning Path Filtering and Alignment for Small Parameter Large Language Models. In 2026 2nd International Conference on Artificial Intelligence and Digital Ethics (ICAIDE) (pp. 29-32). IEEE.

16. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). Concrete problems in AI safety. arXiv preprint arXiv:1606.06565.

17. Russell, S. (2019). Human compatible: Artificial intelligence and the problem of control. Viking.

18. Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 610–623.

19. Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., ... Liang, P. (2021). On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258.

20. Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., ... Dennison, D. (2015). Hidden technical debt in machine learning systems. Advances in Neural Information Processing Systems, 28, 2503–2511.

21. Mittelstadt, B. D., Allo, P., Taddeo, M., Wachter, S., & Floridi, L. (2016). The ethics of algorithms: Mapping the debate. Big Data & Society, 3(2), 1–21.

22. Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 3645–3650.

23. Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., & Meger, D. (2018). Deep reinforcement learning that matters. Proceedings of the AAAI Conference on Artificial Intelligence, 32(1), 3207–3214.

24. Dafoe, A., Hughes, E., Bachrach, Y., Collins, T., McKee, K. R., Leibo, J. Z., ... Graepel, T. (2020). Open problems in cooperative AI. arXiv preprint arXiv:2012.08630.

25. Gabriel, I. (2020). Artificial intelligence, values, and alignment. Minds and Machines, 30, 411–437.