Prompt-Aware Continual Learning for Catastrophic Forgetting Mitigation in Large Language Models

Ranyuan Chen
School of Information Technology, University of Cincinnati, Cincinnati, OH, USA.
chenranyuan@uc.edu

Abstract

Large language models (LLMs) have achieved remarkable generalization across diverse tasks, yet their deployment in dynamically evolving environments exposes a critical vulnerability: catastrophic forgetting when sequentially updated with new knowledge. Traditional fine-tuning overwrites previously acquired capabilities, rendering continuous adaptation impractical for long-lived systems. This paper proposes a prompt-aware continual learning paradigm that mitigates forgetting by treating discrete and continuous prompts as persistent, task-specific memory modules. Rather than modifying model parameters, the backbone remains frozen while a prompt selection mechanism dynamically retrieves contextually appropriate prompt embeddings. We provide a system-level analysis of the architectural trade-offs, infrastructure requirements, and governance implications of this approach. The discussion encompasses prompt storage architectures, retrieval latency, versioning, and load balancing for large-scale deployment. We examine fairness concerns arising from prompt accumulation, particularly the risk of performance degradation for underrepresented tasks and the propagation of prompt-induced biases. Policy dimensions including auditability, ownership of prompt modules, and compliance with emerging AI regulations are addressed. Furthermore, we compare prompt-aware continual learning with rehearsal, regularization, and dynamic expansion strategies, highlighting advantages in modularity and energy efficiency. Robustness to distribution shift, prompt collision, and long-term maintainability are explored through the lens of sustainable model lifecycle management. The paper concludes with forward-looking perspectives on integrating prompt-aware continual learning into multi-tenant AI platforms, emphasizing the need for standardization, monitoring, and inclusive governance frameworks to ensure responsible, equitable, and resilient LLM evolution.

Keywords

continual learning, catastrophic forgetting, large language models, prompt tuning, system architecture, governance, fairness, sustainability.

1. Introduction

Large language models have become foundational infrastructure in modern artificial intelligence systems, underpinning applications that range from conversational agents and code generation to medical summarization and legal reasoning. The prevailing paradigm of pretraining on vast corpora followed by task-specific fine-tuning yields impressive initial performance, but it introduces a fundamental obstacle to the long-term viability of deployed models. When an LLM is sequentially updated with new tasks, domains, or user feedback, its ability to perform previously mastered functions deteriorates sharply, a phenomenon known as catastrophic forgetting. This is not merely an academic curiosity; it has direct operational consequences for systems that must continuously incorporate new knowledge without regressing on earlier capabilities. The conventional remedy of retraining from scratch on all

accumulated data is computationally prohibitive, environmentally unsustainable, and incompatible with privacy constraints that limit access to historical datasets.

Prompt tuning, a parameter-efficient fine-tuning technique, has emerged as a promising vehicle for isolating task-specific adaptations. By prepending a small set of learnable continuous vectors to the input sequence while keeping the entire pretrained model frozen, prompt tuning encapsulates task competence in a compact, modular representation. This modularity suggests a natural alignment with continual learning: if each new task can be mapped to a distinct prompt module, the catastrophic interference that arises from parameter overwriting can be circumvented. However, realizing this vision requires moving beyond static prompt allocation to a prompt-aware continual learning framework in which the system dynamically selects, composes, and manages a growing repository of prompts in response to incoming data streams. The present paper develops a system-level perspective on such a framework, placing equal emphasis on architectural design, infrastructure engineering, governance, fairness, and sustainability. The discussion is intentionally broad, bridging machine learning research, systems engineering, and policy analysis to articulate a comprehensive vision for continual LLM evolution.

2. Background and Related Work

Catastrophic forgetting has been studied extensively since the early days of connectionist modeling, with foundational work observing that sequential training on new tasks disrupts previously learned representations [1]. The continual learning community has since produced a rich taxonomy of mitigation strategies. Regularization-based methods such as elastic weight consolidation penalize changes to parameters deemed important for prior tasks, thereby slowing forgetting at the cost of capacity constraints [2]. Gradient episodic memory and related approaches store a subset of past examples for interleaved replay, trading memory overhead for reduced interference [3]. Generative replay alternatives learn a model of the data distribution to synthesize historical samples, mitigating privacy concerns but introducing additional training complexity [4]. Architectural expansion methods dynamically allocate new parameters for each task, sidestepping interference entirely while inevitably increasing model size. Comprehensive surveys document the strengths and weaknesses of these families, highlighting the trade-off between memory, computation, and plasticity [5].

Concurrently, parameter-efficient tuning methods have revolutionized the adaptation of large pretrained models. Adapters insert small bottleneck modules between transformer layers, achieving competitive performance with minimal trainable parameters [6]. Prefix tuning prepends learnable continuous vectors to the keys and values of the attention mechanism, thereby contextualizing generation without altering the core model [7]. Prompt tuning further simplifies this idea by optimizing only a small set of input embeddings, making task switching as simple as switching prompt vectors [8]. Subsequent work demonstrated that prompt tuning can match full fine-tuning when the model scale is sufficiently large and when prompts are appropriately initialized and positioned [9]. The T5 framework and its variants showed how unified text-to-text architectures benefit from these lightweight adaptation techniques, setting the stage for their incorporation into continual learning pipelines [10].

Large language models such as GPT-3, LLaMA, and BLOOM have demonstrated unprecedented few-shot and zero-shot capabilities, making them ideal candidates for parameter-efficient continual adaptation [11, 12, 13, 14, 15, 16]. Instruction tuning and chain-of-thought prompting have further improved the steerability of such models, underscoring the importance of prompt design as a first-class interface for model behavior [13, 14]. The natural

extension is to maintain a library of prompts that encapsulate distinct competencies. Recent work on selective prompt insertion explores learning when and where to inject prompt tokens into the sequence, enhancing the flexibility and expressiveness of prompt-based adaptation [18]. While these techniques are not explicitly framed as continual learning solutions, they supply the technical substrate for prompt-aware continual learning, in which prompt selection, insertion, and composition become the mechanisms for managing knowledge over time.

3. The Prompt-Aware Continual Learning Paradigm

In the prompt-aware continual learning framework, a frozen pretrained LLM serves as a universal knowledge processor that is never directly updated. Task-specific knowledge is externalized into a growing set of prompt modules, each consisting of a small number of trainable embedding vectors. When a new task arrives, a dedicated prompt is learned using standard prompt tuning optimization, while the backbone remains immutable. Critically, the system incorporates a prompt-aware selector that examines incoming queries and retrieves the most relevant prompt or combination of prompts from the repository. This selector can be implemented as a lightweight neural classifier, a similarity-based retriever, or a learned router that conditions on input embeddings or discrete metadata. During inference, the selected prompt is prepended or interleaved with the input tokens, steering the frozen model toward task-appropriate behavior without any weight modification.

The architectural elegance of this approach lies in its clean separation of concerns. The backbone encapsulates general linguistic and world knowledge acquired during pretraining, while the prompts capture narrow, task-specific procedural knowledge. This separation mirrors long-standing principles in software engineering and cognitive architectures, where stable core systems are extended through pluggable modules. From a continual learning perspective, catastrophic forgetting is eliminated at the parameter level because no parameters are shared across tasks in a conflicting manner. However, forgetting can still manifest at the prompt selection level: if the selector is updated using only recent task data, it may lose the ability to correctly retrieve older prompts, effectively rendering them inaccessible. Mitigating this form of selector drift requires careful calibration of the selector's training regimen, potentially incorporating rehearsed query-prompt associations or generative replay of past task distributions to keep the selection policy stable.

The prompt capacity and the granularity of prompt assignment represent critical design axes. A single prompt per task is the simplest configuration, but it fails to exploit cross-task commonalities and may lead to prompt bloat when dozens or hundreds of tasks accumulate. Alternatively, prompts can be shared, composed, or factorized into reusable sub-prompts that correspond to specific skills such as summarization, translation tone, or entity extraction. This compositional design reduces storage overhead and can improve generalization, but it introduces the challenge of learning composable representations that do not interfere. The selector must then solve a more complex combinatorial retrieval problem, which may require hierarchical indexing or attention-based prompt fusion mechanisms. The trade-off between specialization and sharing directly impacts the long-term scalability and efficiency of the system.

Comparisons with other architectural expansion methods reveal distinctive advantages. In dynamic network expansion, entire parameter blocks are added, which quickly inflates the model footprint and complicates deployment on resource-constrained edge devices. Prompt modules, by contrast, occupy orders of magnitude less space; a typical prompt of length 100 with hidden dimension 4096 comprises only 400,000 floating-point numbers, a fraction of a

percent of the backbone's size. Moreover, prompts can be exchanged without reloading the entire model, enabling rapid task switching with minimal memory bandwidth. This compactness makes prompt-aware continual learning especially suitable for federated learning scenarios, where prompts are trained on-device and aggregated centrally without exposing raw data or full model weights.

4. System Architecture and Infrastructure Design

Operationalizing prompt-aware continual learning at scale demands a purpose-built infrastructure that extends beyond a simple lookup table. The prompt repository functions as a distributed key-value store where each prompt is indexed by a task identifier, a semantic hash of representative queries, or an embedding of the task's description. A critical infrastructure challenge is maintaining low-latency prompt retrieval within the tight inference budgets typical of production LLM services. Caching strategies that keep the most frequently used prompts in accelerator memory and prefetching mechanisms that anticipate upcoming task context based on session continuity can mitigate retrieval overhead. Consistency protocols must ensure that prompt updates propagated from continuous training loops do not disrupt ongoing inference, necessitating versioned prompt stores and atomic swap operations.

The storage footprint of prompt collections grows linearly with the number of tasks, which may reach thousands in multi-tenant platforms serving diverse enterprise clients. While individual prompts are small, the cumulative storage can become nontrivial when replicated across geographically distributed inference clusters for fault tolerance and latency reduction. Hierarchical prompt management policies can alleviate this pressure by consolidating rarely used prompts in lower-cost object storage and promoting active prompts to high-performance memory tiers. Lifecycle management protocols should define criteria for prompt deprecation or merging when tasks become obsolete or when multiple prompts exhibit high functional overlap, thus preventing unbounded growth.

Versioning, auditing, and rollback capabilities are indispensable for trustworthy deployment. Every prompt update must be logged with metadata that includes the training dataset fingerprint, optimization hyperparameters, responsible team, and performance metrics on held-out validation sets. This audit trail supports forensic analysis when a prompt-induced regression is detected and enables rapid rollback to a known-good version without affecting the frozen backbone. Such infrastructure aligns with MLOps best practices and is increasingly mandated by regulatory frameworks that require transparency and accountability for automated decision-making systems. Integrating prompt-aware continual learning into existing CI/CD pipelines requires extending model registries to accommodate prompt artifacts alongside full model checkpoints, with standardized evaluation harnesses that assess both individual task performance and cross-task interference.

Load balancing across inference accelerators introduces additional complexity when prompts are dynamic. Traditional batching strategies assume a homogeneous processing context; however, different prompts may steer attention patterns and generation trajectories differently, leading to variable computational costs. The infrastructure must be capable of profiling the latency profile of prompt-model combinations and scheduling batches that minimize tail latency while maximizing throughput. This requirement suggests a co-design of prompt insertion mechanisms with hardware-aware compilation toolchains that can optimize kernel launches based on the specific prompt tokens, potentially applying operator fusion or speculative decoding techniques that are prompt-aware.

5. Governance, Fairness, and Policy Implications

The shift from monolithic model updates to modular prompt management transforms the governance landscape. Prompts can be viewed as executable policy artifacts that encode not only linguistic competence but also normative stances, stylistic biases, and sensitivity to protected attributes. As a prompt repository grows, so does the risk that certain tasks—particularly those serving marginalized communities or representing low-resource languages—receive insufficient maintenance and suffer from gradual performance erosion, a phenomenon analogous to representation forgetting. This fairness concern is compounded when the prompt selector is optimized for aggregate accuracy, leading to a "tyranny of the majority" where high-frequency tasks dominate selection training and under-served tasks become gradually inaccessible.

Mitigating these risks requires embedding fairness constraints directly into the continual learning governance framework. Prompt selection policies should be audited regularly for equitable retrieval rates across task categories, and corrective mechanisms such as stratified selector fine-tuning or minimum retrieval frequency quotas should be enforced. Furthermore, the provenance of prompts must be transparent: who designed a prompt, on whose data it was trained, and under what ethical guidelines. In multi-stakeholder environments where third-party developers contribute prompts, a certification process akin to app store reviews can help prevent malicious or biased prompt injection. The European Union's AI Act and similar emerging regulations increasingly require such documentation and oversight for high-risk AI systems, making prompt governance not merely a best practice but a compliance necessity.

The continual nature of prompt updates also raises questions about liability and version accountability. When an LLM deployed in a clinical decision-support context produces an erroneous recommendation due to a recently added prompt, determining whether the fault lies with the prompt, the selector, the backbone, or their interaction is nontrivial. A robust governance framework must define clear ownership boundaries, incident response protocols, and compensation mechanisms. Public registries of prompt versions with attested performance benchmarks could become standard practice, enabling independent verification and fostering trust among users, regulators, and civil society organizations. The concept of prompt-as-a-service introduces economic dimensions in which prompts are treated as tradeable intellectual property, necessitating licensing models, royalty structures, and antitrust considerations to prevent monopolistic control over critical language capabilities.

6. Deployment Strategies and Sustainability

Deploying prompt-aware continual learning in production environments demands integration with orchestration platforms that support gradual rollout, experimentation, and automatic rollback. Canary deployments, where new prompts are first served to a small fraction of traffic while monitoring key performance indicators, reduce the blast radius of regressions. A/B testing frameworks adapted for prompt variants enable data-driven decisions about which prompt version best balances accuracy, fairness, and user satisfaction. Because the backbone model remains unchanged, these experiments can be conducted with drastically lower storage and bandwidth costs compared to shipping full model replicas, lowering the barrier for continuous improvement and community-driven innovation.

Sustainability considerations are central to the value proposition of prompt-aware continual learning. Training a large language model from scratch consumes hundreds of megawatt-hours of electricity and generates substantial carbon emissions. Full fine-tuning for each new

task multiplies this cost, making continuous retraining environmentally untenable. Prompt tuning, by contrast, involves optimizing a few thousand parameters, typically requiring less than 1 percent of the compute of full fine-tuning. The resulting prompt modules can be shared, reused, and composed, amplifying their efficiency gains over the model lifecycle. Nevertheless, one must account for the additional inference cost of prompt retrieval, the energy footprint of the prompt repository's storage infrastructure, and the overhead of selector computation. A holistic life-cycle assessment across hardware generations and electricity grid carbon intensities is needed to validate the net environmental benefit.

Long-term maintainability also depends on the ecosystem's ability to handle prompt decay and obsolescence. As the underlying language and world knowledge evolve, tasks that were well-served by older prompts may require adaptation. Rather than discarding prompts, one can apply lightweight fine-tuning of prompts on fresh data, a process that is orders of magnitude cheaper than retraining the backbone. Moreover, prompts can be transferred across model generations; when a backbone is upgraded, prompts trained on the previous version can serve as a strong initialization for the new model, preserving institutional knowledge. This forward compatibility creates a compelling economic case for enterprises to invest in prompt libraries as durable assets that outlast individual model releases.

7. Robustness and Long-Term Adaptability

The robustness of prompt-aware continual learning systems is threatened by several failure modes. Prompt collision occurs when two contexts map to the same prompt or when prompts interfere destructively during composition, producing nonsensical or harmful outputs. Robust prompt selection demands robust out-of-distribution detection so that unfamiliar inputs do not silently trigger inappropriate prompts. Integrating uncertainty quantification into the selector, such as by modeling a reject option that falls back to a general-purpose prompt or defers to human oversight, can contain the damage. Adversarial robustness must also be considered: maliciously crafted inputs could attempt to trigger sensitive prompts, making prompt access control and input sanitization imperative.

Distribution shift over time poses a subtler challenge. As user behavior, language conventions, and factual knowledge evolve, the frozen backbone's pretrained representations may become stale, and even well-tuned prompts may lose effectiveness. Continual learning systems must therefore incorporate mechanisms for monitoring distribution shift and triggering prompt retraining or backbone refreshing when drift exceeds predefined thresholds. Meta-learning strategies that train the prompt selector to quickly adapt to shifting distributions, perhaps by maintaining a small buffer of recent examples, can enhance responsiveness without compromising long-term stability. The long-term vision is a self-sustaining ecosystem in which the model's knowledge is continuously curated through prompt updates, with human-in-the-loop oversight reserved for ambiguous or high-stakes decisions.

Cross-domain comparisons highlight that prompt-aware continual learning is not a panacea but a complementary tool. In safety-critical domains such as aviation or nuclear power, the rigidity of frozen backbones may be valued over the flexibility of prompt switching. In such contexts, formal verification of the backbone combined with strictly versioned and tested prompt modules could provide a balanced approach. For consumer-facing applications with rapidly changing content moderation policies, the ability to instantly deploy a new prompt that updates the model's acceptable use boundaries without retraining offers significant operational agility. The success of these deployments ultimately rests on robust testing ecosystems that simulate long-horizon continual learning scenarios with evolving task

streams, adversarial interventions, and hardware faults, enabling engineers to assess system resilience before real-world rollouts.

8. Conclusion

Prompt-aware continual learning offers a compelling architectural paradigm for managing catastrophic forgetting in large language models while enabling efficient, modular, and auditable adaptation over time. By freezing the backbone and externalizing task-specific knowledge into compact, retrievable prompt modules, the framework decouples stability from plasticity, reduces computational overhead, and aligns with modern software engineering principles of modularity and separation of concerns. This paper has examined the system-level implications of such an approach, spanning infrastructure design, governance, fairness, sustainability, and robustness. The analysis reveals that while prompt-aware continual learning can drastically reduce the cost and complexity of maintaining ever-evolving language models, it introduces new challenges around selector drift, prompt version management, equitable access, and long-term resource management. Addressing these challenges will require close collaboration between machine learning researchers, systems engineers, policymakers, and domain stakeholders to build standardized prompt registries, fairness auditing protocols, and sustainable lifecycle management practices. Future research should explore composable prompt representations, uncertainty-aware selectors, and automated prompt evolution strategies that can keep pace with the accelerating dynamics of language, culture, and knowledge. As LLMs become increasingly embedded in societal infrastructure, the principles of prompt-aware continual learning will likely prove indispensable for ensuring that these systems remain accurate, fair, and accountable across their entire operational lifespan.

References

1. French, R. M. (1999). Catastrophic forgetting in connectionist networks. Trends in Cognitive Sciences, 3(4), 128–135.
2. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., ... & Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. Proceedings of the National Academy of Sciences, 114(13), 3521–3526.
3. Lopez-Paz, D., & Ranzato, M. A. (2017). Gradient episodic memory for continual learning. In Advances in Neural Information Processing Systems (pp. 6467–6476).
4. Shin, H., Lee, J. K., Kim, J., & Kim, J. (2017). Continual learning with deep generative replay. In Advances in Neural Information Processing Systems (pp. 2990–2999).
5. Parisi, G. I., Kemker, R., Part, J. L., Kanan, C., & Wermter, S. (2019). Continual lifelong learning with neural networks: A review. Neural Networks, 113, 54–71.
6. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., de Laroussilhe, Q., Gesmundo, A., ... & Gelly, S. (2019). Parameter-efficient transfer learning for NLP. In Proceedings of the 36th International Conference on Machine Learning (pp. 2790–2799).
7. Li, X. L., & Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (pp. 4582–4597).

8. Lester, B., Al-Rfou, R., & Constant, N. (2021). The power of scale for parameter-efficient prompt tuning. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (pp. 3045–3059).

9. Liu, X., Ji, K., Fu, Y., Du, Z., Yang, Z., & Tang, J. (2022). P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (pp. 2941–2953).

10. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21(140), 1–67.

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

12. 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.

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

14. 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. In Advances in Neural Information Processing Systems (pp. 24824–24837).

15. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M. A., Lacroix, T., ... & Lample, G. (2023). LLaMA: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971.

16. Scao, T. L., Fan, A., Akiki, C., Pavlick, E., Ilić, S., Hesslow, D., ... & Wolf, T. (2022). BLOOM: A 176B-parameter open-access multilingual language model. arXiv preprint arXiv:2211.05100.

17. Wang, Y., Mishra, S., Alipoormolabashi, P., Kordi, Y., Mirzaei, A., Arunkumar, A., ... & Khashabi, D. (2022). Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (pp. 5085–5109).

18. Zhu, W., & Tan, M. (2023, December). SPT: Learning to selectively insert prompts for better prompt tuning. In Proceedings of the 2023 conference on empirical methods in natural language processing (pp. 11862-11878).

19. Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., ... & Zhang, Y. (2023). Sparks of artificial general intelligence: Early experiments with GPT-4. arXiv preprint arXiv:2303.12712.

20. Biderman, S., Schoelkopf, H., Anthony, Q., Bradley, H., O'Brien, K., Hallahan, E., ... & Gao, L. (2023). Pythia: A suite for analyzing large language models across training and scaling. In Proceedings of the 40th International Conference on Machine Learning (pp. 2397–2430).

21. Rolnick, D., Ahuja, A., Schwarz, J., Lillicrap, T. P., & Wayne, G. (2019). Experience replay for continual learning. In Advances in Neural Information Processing Systems (pp. 350–360).

22. van de Ven, G. M., & Tolias, A. S. (2019). Three scenarios for continual learning. arXiv preprint arXiv:1904.07734.

23. Madaan, A., Tandon, N., Clark, P., & Yang, Y. (2022). Self-refine: Iterative refinement with self-feedback. In Advances in Neural Information Processing Systems (pp. 11711–11731).

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

25. Wu, T., Luo, M., & Xu, X. (2023). Continual learning with prompts: Is that possible? In Findings of the Association for Computational Linguistics: ACL 2023 (pp. 8431–8445).