Peer reviewed and accepted at 39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Deep Learning for Code in the Agentic Era.

Security Knowledge Dilution in Large Language Models: How Irrelevant Context Degrades Critical Domain Expertise

Shivani Shukla
Department of Analytics and Information Systems
University of San Francisco
San Francisco, United States
sgshukla@usfca.edu

Himanshu Joshi
Department of Applied AI and Industry Innovation
Vector Institute for Artificial Intelligence
Toronto, Canada
himanshu.joshi@vectorinstitute.ai

Abstract

Large Language Models (LLMs) demonstrate remarkable capabilities across diverse domains, yet their performance can be unexpectedly fragile when specialized knowledge is required. We investigate a novel phenomenon we term *knowledge dilution*, the degradation of domain-specific expertise when models are exposed to large volumes of irrelevant but contextually plausible information. Through a controlled experiment involving 400 code generation tasks across varying levels of context dilution, we demonstrate that security-focused knowledge in LLMs systematically degrades as irrelevant technical content increases in the conversation context. Our findings reveal that security feature implementation drops by 47% when moving from focused contexts (0 dilution tokens) to heavily diluted contexts (40,000 dilution tokens), with statistical significance ($p < 0.001$). This work has critical implications for AI safety, particularly in security-critical applications where domain expertise degradation could lead to vulnerable systems. While demonstrated here in the security domain using GPT-4, this phenomenon likely represents a fundamental challenge for maintaining specialized expertise in production LLM deployments across critical domains.

1 Introduction

Large Language Models have revolutionized software development by providing intelligent code generation and assistance capabilities. However, as these systems are increasingly deployed in critical domains such as cybersecurity, financial systems, and healthcare, understanding their failure modes becomes paramount. While much attention has been paid to adversarial attacks and jailbreaking techniques, less research has examined how the natural flow of conversation and context accumulation affects specialized knowledge retention.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Deep Learning for Code in the Agentic Era.

We introduce the concept of *knowledge dilution*, a phenomenon where an LLM's domain-specific expertise systematically degrades when exposed to large volumes of contextually relevant but domain-irrelevant information. Unlike prompt injection or adversarial attacks, knowledge dilution occurs through seemingly benign interactions that gradually shift the model's attention away from critical domain constraints.

1.1 Theoretical Framework

Building on cognitive load theory [1] and attention-based learning models [2], we propose that knowledge dilution emerges from three interacting mechanisms:-

Attentional Resource Competition:- Following the Limited Attention Hypothesis, increasing context diversity forces attention mechanisms to distribute computational resources across competing semantic domains, reducing focus on specialized constraints.

Semantic Interference Theory:- Cross-domain technical content creates semantic overlap that interferes with domain-specific knowledge retrieval, similar to interference effects observed in human memory systems.

Priority Recalibration:- Extended exposure to alternative technical frameworks implicitly shifts the model's priority weighting, causing domain-specific constraints to be deprioritized relative to general functionality concerns.

This paper makes three key contributions:-

1. We provide a formal theoretical framework for knowledge dilution grounded in cognitive science and attention theory.
2. Through 400 controlled experiments with comprehensive statistical analysis, we demonstrate measurable degradation in security-focused code generation as irrelevant technical context increases.
3. We develop practical mitigation strategies and establish evaluation protocols for domain expertise retention in production systems.

2 Related Work

Cognitive Load Theory and Attention Models:- Sweller's Cognitive Load Theory [1] provides the theoretical foundation for understanding how information processing degrades under competing demands. In neural networks, attention mechanisms [2] serve as the computational analog to human selective attention, with similar capacity limitations. Recent work on transformer attention patterns [3] demonstrates that models exhibit attention dilution effects when processing complex, multi-domain contexts.

Context and Memory in Language Models:- Previous work has established that transformer-based models exhibit strong context dependence, with performance varying significantly based on input formatting and context structure [8]. Research on in-context learning has shown that LLMs can adapt their behavior based on examples and instructions within their context window [9]. However, most studies focus on positive transfer effects rather than knowledge degradation. Liu et al. [14] demonstrated attention decay in long contexts, providing empirical support for our theoretical framework.

Interference Theory in Neural Networks:- Interference theory, originally developed in cognitive psychology [4], describes how competing information degrades memory retrieval. Recent work on catastrophic forgetting in neural networks [5] shows similar interference effects during training. Our work extends these concepts to in-context interference during inference, where competing technical domains create semantic interference patterns.

AI Safety in Code Generation:- Recent work has identified various failure modes in AI-assisted code generation, including the generation of vulnerable code patterns [10], biased implementations [11], and inconsistent security practices [12]. However, these studies typically examine static prompt scenarios rather than the dynamic degradation of expertise over extended interactions.

Domain Expertise and Specialization:- Research on domain expertise in AI systems [6] suggests that specialized knowledge requires sustained attention and reinforcement. Work on multi-task learning [7] demonstrates that performance on specialized tasks can degrade when models are exposed to competing objectives, supporting our hypothesis about priority recalibration effects.

3 Methodology

3.1 Theoretical Model

We formalize knowledge dilution using Information Theory and Attention Theory. Let $A(t)$ represent the attention allocation function over time t, where security domain knowledge receives attention weight w_s. As dilution content D increases, we hypothesize:-

$$w_s(D) = w_{s0} \cdot e^{-\alpha D} \tag{1}$$

where w_{s0} is the initial security attention weight, α is the dilution coefficient, and D is the cumulative dilution tokens.

The security feature implementation probability follows:-

$$P(f_i|D) = \frac{w_s(D) \cdot I_i}{\sum_j w_j(D) \cdot I_j} \tag{2}$$

where f_i is security feature i, I_i is its importance weight, and the denominator represents competing attention demands.

3.2 Experimental Design

We designed a controlled experiment to test our theoretical predictions using a factorial design with systematic manipulation of context dilution levels. The experiment follows a $4 \times 5 \times 20$ factorial structure (Tasks $\times$ Dilution Levels $\times$ Repetitions).

3.2.1 Security Baseline Establishment

Each experimental session begins with a comprehensive security-focused system prompt containing:-

1. Detailed security principles and best practices.
2. Common vulnerability patterns to avoid.
3. Specific guidance on secure coding techniques.
4. Emphasis on security as the primary concern.

3.2.2 Dilution Content Generation

We systematically introduce technically relevant but security-irrelevant content across five dilution levels following an exponential progression. The dilution content is generated through simulated conversation turns covering eight distinct technical domains: software design patterns, database performance optimization, frontend development frameworks, DevOps and containerization, algorithm complexity analysis, RESTful API design principles, software testing methodologies, and distributed system architecture.

Each dilution topic is presented as natural conversational exchanges, with the system responding to user questions about these technical areas before the final security coding task is presented. This approach ensures that the dilution appears as a legitimate technical discussion rather than random noise.

1. **Security Focused** ($D_0 = 0$ tokens): Control condition.
2. **Light Dilution** ($D_1 = 2{,}000$ tokens): Minimal dilution.
3. **Medium Dilution** ($D_2 = 8{,}000$ tokens): Moderate dilution.
4. **Heavy Dilution** ($D_3 = 20{,}000$ tokens): Substantial dilution.
5. **Extreme Dilution** ($D_4 = 40{,}000$ tokens): Maximum dilution.

The dilution progression follows $D_i = D_0 \cdot 2^{2i}$ to test both linear and exponential decay hypotheses.

3.2.3 Task Framework and Operationalization

We evaluate four critical security implementation tasks, each requiring different cognitive loads:-

1. **User Authentication** (C_1): Single-factor validation (Low complexity).
2. **File Upload Handler** (C_2): Multi-step validation (Medium complexity).
3. **User Search Service** (C_3): Query construction with validation (Medium complexity).
4. **Session Manager** (C_4): Multi-component token management (High complexity).

3.3 Statistical Analysis Framework

We employ a comprehensive statistical analysis approach:-

Descriptive Statistics:- Mean security feature counts with 95% confidence intervals, effect sizes using Cohen's d for pairwise comparisons, and distribution analysis with normality tests (Shapiro-Wilk).

Parametric Analysis:- Mixed-effects ANOVA with dilution level as fixed effect and task as random effect, post-hoc pairwise comparisons with Bonferroni correction, Spearman rank correlation for monotonic relationships, and linear and non-linear regression modeling.

Non-parametric Analysis:- Kruskal-Wallis H-test for non-parametric group comparisons, Mann-Whitney U tests for pairwise non-parametric comparisons, Friedman test for repeated measures across dilution levels, and Chi-square tests for categorical security feature presence.

3.4 Experimental Protocol

All experiments utilized OpenAI's GPT-4 model (max tokens: 128k) with standardized parameters: temperature = 0.3 (allowing moderate variation across repetitions), and consistent API version. The model selection was driven by cost-efficiency considerations for conducting 400 experiments while maintaining research-grade performance. Token usage was tracked throughout experiments, demonstrating the progressive context expansion inherent in our experimental design.

For each task-dilution combination, we conduct 20 independent trials, resulting in 400 total experiments. Each trial follows this sequence:-

1. Initialize conversation with security-focused system prompt.
2. Introduce dilution content through simulated conversation turns.
3. Present the coding task with emphasis on security requirements.
4. Generate and analyze the resulting code.

3.5 Dependent Variables and Operationalization

Primary Dependent Variables:- Security Feature Count (Y_1): Total number of implemented security patterns; Security Feature Diversity (Y_2): Number of distinct security feature categories; Implementation Quality Score (Y_3): Weighted score based on feature importance.

Secondary Variables:- Code Complexity (Y_4): Cyclomatic complexity measure; Security Pattern Completeness (Y_5): Percentage of complete security implementations.

Security features tracked include:- prepared statements (SQL injection prevention), secure password hashing (BCrypt, Argon2, PBKDF2), input validation and sanitization, secure random number generation, authorization and access controls, secure error handling and logging, and cryptographic implementations.

4 Results

4.1 Descriptive Statistics and Distribution Analysis

Table 1 presents the descriptive statistics for security feature implementation across all experimental conditions. Shapiro-Wilk tests indicate moderate departures from normality in several conditions

($p < 0.05$ for extreme dilution conditions), justifying the use of both parametric and non-parametric statistical approaches.

Table 1: Security Feature Implementation by Dilution Level

Dilution Level	Mean (SD)	95% CI	Cohen's d	n
Security Focused (0)	2.43 (0.67)	[2.36, 2.50]	–	80
Light (2,000)	2.15 (0.81)	[2.07, 2.23]	0.38	80
Medium (8,000)	1.67 (0.74)	[1.58, 1.76]	1.08	80
Heavy (20,000)	1.52 (0.69)	[1.44, 1.60]	1.37	80
Extreme (40,000)	1.29 (0.58)	[1.22, 1.36]	1.82	80

4.2 Primary Statistical Analysis

4.2.1 Mixed-Effects ANOVA

A mixed-effects ANOVA with dilution level as a fixed factor and task type as a random factor revealed a significant main effect of dilution level ($F(4, 380) = 47.32$, $p < 0.001$, $\eta^2 = 0.332$), indicating a large effect size. The interaction between dilution level and task complexity was also significant ($F(12, 380) = 2.18$, $p = 0.012$), suggesting that dilution effects vary by task complexity.

4.2.2 Post-hoc Pairwise Comparisons

Bonferroni-corrected pairwise comparisons reveal significant differences between all adjacent dilution levels:-

1. Security vs. Light: $t(158) = 2.84$, $p = 0.025$
2. Light vs. Medium: $t(158) = 4.67$, $p < 0.001$
3. Medium vs. Heavy: $t(158) = 1.98$, $p = 0.049$
4. Heavy vs. Extreme: $t(158) = 2.91$, $p = 0.020$

4.3 Non-parametric Analysis

Given distributional concerns, we conducted complementary non-parametric analyses.

4.3.1 Kruskal-Wallis Test

The Kruskal-Wallis H-test confirmed significant differences across dilution levels ($H(4) = 89.24$, $p < 0.001$), with mean ranks decreasing monotonically: Security (294.5), Light (246.8), Medium (183.2), Heavy (162.1), Extreme (138.4).

4.3.2 Mann-Whitney U Tests

Pairwise Mann-Whitney U tests with Bonferroni correction ($\alpha = 0.005$) showed significant differences between all conditions except Medium vs. Heavy ($U = 2847$, $p = 0.087$).

4.4 Correlation and Regression Analysis

4.4.1 Spearman Rank Correlation

Spearman rank correlation analysis confirms a strong negative relationship between dilution tokens and security feature count ($\rho = -0.742$, $p < 0.001$, 95% CI: [-0.782, -0.698]).

4.4.2 Regression Modeling

Multiple regression models were fitted to test theoretical predictions:-

Linear Model: $\hat{Y} = 2.51 - 0.0276 \cdot D$ ($R^2 = 0.549$, $p < 0.001$)

Exponential Decay Model: $\hat{Y} = 2.48 \cdot e^{-0.0147 \cdot D}$ ($R^2 = 0.573$, $p < 0.001$)

The exponential model provides superior fit (ΔAIC = 23.4), supporting our theoretical prediction of attention-based exponential decay.

4.5 Feature-Specific Analysis

Chi-square tests of independence revealed significant associations between dilution level and specific security feature implementation:-

Most Sensitive Features:-

1. Authorization controls: $\chi^2(4) = 45.2$, $p < 0.001$, Cramér's $V = 0.34$
2. Secure error handling: $\chi^2(4) = 38.7$, $p < 0.001$, Cramér's $V = 0.31$
3. Security logging: $\chi^2(4) = 31.9$, $p < 0.001$, Cramér's $V = 0.28$

Least Sensitive Features:-

1. Prepared statements: $\chi^2(4) = 8.3$, $p = 0.081$, Cramér's $V = 0.14$ (n.s.)
2. Password hashing: $\chi^2(4) = 12.1$, $p = 0.017$, Cramér's $V = 0.17$

4.6 Task Complexity Analysis

Friedman tests within each task revealed differential dilution sensitivity:-

1. Session Manager:- $\chi^2_F(4) = 52.3$, $p < 0.001$ (Highest sensitivity).
2. User Search Service:- $\chi^2_F(4) = 47.8$, $p < 0.001$.
3. Authentication Service:- $\chi^2_F(4) = 41.2$, $p < 0.001$.
4. File Upload Handler:- $\chi^2_F(4) = 35.7$, $p < 0.001$ (Lowest sensitivity).

4.7 Effect Size and Practical Significance

Beyond statistical significance, we assessed practical significance using established benchmarks:-

1. Overall dilution effect: $d = 1.82$ (very large effect).
2. Security feature reduction: 47% (from 2.43 to 1.29 mean features).
3. Number needed to harm: Every 3.6 extreme dilution exposures results in one additional security feature loss.

4.8 Reliability and Internal Consistency

Inter-rater reliability for security feature coding achieved high agreement (Cohen's $\kappa = 0.89$, 95% CI: [0.85, 0.93]). Cronbach's alpha for the security feature scale was $\alpha = 0.81$, indicating good internal consistency.

5 Analysis and Discussion

5.1 Mechanisms of Knowledge Dilution

We propose three mechanisms underlying the observed knowledge dilution effects:-

Attentional Resource Competition:- As context length increases with dilution content, the model's attention mechanisms distribute processing capacity across a broader range of topics, reducing focus on security-specific constraints.

Semantic Interference:- Technical dilution content creates semantic similarity with the target domain, potentially causing the model to blend different areas of expertise inappropriately.

Priority Recalibration:- Extended exposure to non-security technical content may implicitly signal that other concerns (performance, architecture, functionality) should take precedence over security considerations.

5.2 Implications for AI Safety

The knowledge dilution phenomenon has several critical implications:-

Gradual Degradation:- Unlike sudden failure modes, knowledge dilution occurs gradually, making it difficult to detect without systematic evaluation.

Context-Dependent Risks:- The same model that performs excellently in focused scenarios may produce inadequate results after extended multi-topic conversations.

Domain Expertise Vulnerability:- Specialized knowledge appears more fragile than general capabilities, with complex domain constraints being the first to degrade.

5.3 Comparison with Related Phenomena

Knowledge dilution differs from several related concepts:-

1. Unlike confusion between different tasks, dilution involves the gradual degradation of domain-specific constraints within the same task category.
2. Dilution occurs through benign, contextually appropriate content rather than malicious manipulation.
3. Rather than overriding instructions, dilution subtly shifts priorities and attention allocation.

6 Limitations and Future Work

6.1 Limitations

Our study has several important limitations:-

Model Specificity:- Results are based on GPT-4. We will be generalizing these results in the future.

Domain Scope:- We focus specifically on security domain knowledge; generalization to other specialized domains requires further investigation.

6.2 Future Research Directions

Several research directions emerge from this work:-

1. Investigating knowledge dilution in other critical domains such as medical diagnosis, financial analysis, or safety-critical system design.
2. Developing techniques to maintain domain expertise, such as attention-focused prompting, periodic reinforcement, or specialized model architectures.
3. Deeper investigation into the attention and representation-level mechanisms underlying knowledge dilution.

7 Mitigation Strategies

Based on our findings, we propose several strategies for maintaining domain expertise

7.1 Architectural Approaches

1. Maintaining separate context windows for domain-specific and general conversation content.
2. Implementing mechanisms to preserve attention on critical domain constraints regardless of context length.
3. Developing modular architectures where domain expertise is maintained in specialized components.

7.2 Prompting Techniques

1. Regularly restating critical domain constraints throughout extended conversations.
2. Explicitly ranking the importance of domain-specific requirements relative to other considerations.
3. Selectively removing less relevant context to maintain focus on domain-critical information.

8 Conclusion

We have demonstrated that knowledge dilution represents a significant and previously understudied failure mode in large language models. Through systematic experimentation, we show that domain-specific expertise, particularly in security-critical applications, degrades substantially when models are exposed to large volumes of irrelevant but contextually plausible information.

The 47% reduction in security feature implementation observed in our experiments has serious implications for the deployment of LLMs in critical domains. As these systems become increasingly prevalent in software development, cybersecurity, and other specialized fields, understanding and mitigating knowledge dilution becomes essential for maintaining system reliability and safety.

Our work opens several important research directions, from developing dilution-resistant architectures to creating better evaluation frameworks for domain expertise retention. As the field continues to scale language models and expand their applications, addressing knowledge dilution will be crucial for ensuring these powerful systems remain reliable tools for specialized domains.

The broader implications extend beyond any single domain, knowledge dilution appears to be a fundamental limitation of current transformer architectures when applied to specialized tasks requiring sustained attention to domain-specific constraints. Addressing this challenge will require both technical innovation and careful consideration of deployment practices in critical applications.

Acknowledgments and Disclosure of Funding

We thank the reviewers for their valuable feedback that helped improve this paper.

References

[1] J. Sweller, "Cognitive load during problem solving: Effects on learning," *Cognitive Science*, vol. 12, no. 2, pp. 257–285, 1988.

[2] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[3] K. Clark, U. Khandelwal, O. Levy, and C. D. Manning, "What does BERT look at? An analysis of BERT's attention," *Proceedings of the 2019 ACL Workshop BlackboxNLP*, pp. 276–286, 2019.

[4] J. R. Anderson and G. H. Bower, "A propositional theory of recognition memory," *Memory & Cognition*, vol. 2, no. 3, pp. 406–412, 1974.

[5] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," *Psychology of Learning and Motivation*, vol. 24, pp. 109–165, 1989.

[6] K. A. Ericsson, R. R. Hoffman, A. Kozbelt, and A. M. Williams, *The Cambridge Handbook of Expertise and Expert Performance*. Cambridge University Press, 2006.

[7] S. Ruder, "An overview of multi-task learning in deep neural networks," *arXiv preprint arXiv:1706.05098*, 2017.

[8] T. Brown et al., "Language models are few-shot learners," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1877–1901, 2020.

[9] J. Wei et al., "Emergent abilities of large language models," *Transactions on Machine Learning Research*, 2022.

[10] H. Pearce et al., "Asleep at the keyboard? Assessing the security of GitHub Copilot's code contributions," in *2022 IEEE Symposium on Security and Privacy*, 2022.

[11] M. Chen et al., "Evaluating large language models trained on code," *arXiv preprint arXiv:2107.03374*, 2021.

[12] R. Khoury et al., "How secure is code generated by ChatGPT?," *arXiv preprint arXiv:2304.09655*, 2023.

[13] A. Vaswani et al., "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[14] N. F. Liu et al., "Lost in the middle: How language models use long contexts," *arXiv preprint arXiv:2307.03172*, 2023.