Adaptive Context-Aware Multi-Layer Prompt Security and Output Governance Framework for Secure Large Language Model Deployment

Shakthi N V
Information Science and Engineering
BMS Institute of Technology and Management
Bengaluru, India
1by23is194@bmsit.in

Sowrabhi Narayanan
Information Science and Engineering
BMS Institute of Technology and Management
Bengaluru, India
1by23is214@bmsit.in

Vineeth Sagar H L
Information Science and Engineering
BMS Institute of Technology and Management
Bengaluru, India
1by23is250@bmsit.in

M Sai Vikram
Information Science and Engineering
BMS Institute of Technology and Management
Bengaluru, India
1by23is262@bmsit.in

Sonnegowda K
Information Science and Engineering
BMS Institute of Technology and Management
Bengaluru, India
sonnegowda17@bmsit.in

***Abstract* — Background: Enterprise applications, systems and platforms are increasingly using Large Language Models (LLMs). However, with their growing adoption, they've also given rise to major security risks including prompt injections, attacks and unintentional information leaks. This highlights a need for more robust safety controls Problem Identified: Existing safety controls are based mainly on simple keyword blocking or single-stage moderation, focusing on prompts in isolation. These mechanisms are not effective in identifying attacks, multistage attacks and attacks in multimodal content like images or audio. Goals: The purpose of this project is to design and develop an intelligent and contextual-based multi-layered security system capable of detecting, preventing and removing prompt-based attacks and enable safe and reliable interactions with LLM. Methods: We use a multilayer pinned detection system that evaluates the text lexically, semantically, contextually and behaviorally. It also has the capability to process text, audio and images and post-generation processing such as verification via policy and sandbox. Results: Our system improves the ability to detect less direct and multi-step attacks, and to minimize attacks by producing safe prompts and safe responses. It also ensure explainable monitoring through audit logs.**

Keywords: LLMSecurity, Prompt Injection, Multi-Layer Design, Contextual Analysis, Multi-Modal Processing, Output Control, Explainable AI

I. INTRODUCTION

A. Background and Motivation

Large Language Models (LLMs) are becoming increasingly important in AI systems such as chatbots, content creation, decision-making, and other applications. However, they are vulnerable to prompt injection vulnerabilities, which allow users to tamper, evade rules, exfiltrate data, or manipulate model behavior. This is especially worrying when LLMs are used in systems handling sensitive data or in autonomous systems. Existing control measures primarily involve word-based security based on isolated prompts. These approaches are often context ignorant and have difficulty with subtle or multimodal attacks on text, image and audio inputs.

This project will explore a context-aware, multi-vector approach to security. Our framework examines prompts from multiple dimensions - keywords, interaction history, and conversation history - before LLM processing. Through the inclusion of a semantic analysis component, the system can more accurately discern between users' genuine curiosity and potential malicious intent expressed through more complex wording. Additionally, the framework includes multi-modal confirmation to confirm that prompts are not disguised in non-verbal communications, like metadata or depth clues in an image. This comprehensive monitoring also includes the dynamics of the dialogue, ensuring that a user cannot "salami-slice" a conversation, building a malicious payload over several turns. It also includes screening layers to ensure minimal impact on performance, with processing only occurring if malicious activity is detected. In conclusion, this work offers a template for an adaptive monitoring framework that can adapt to new kinds of jailbreak attacks. This work bridges the gap between shallow content screening and in-depth semantic comprehension to achieve helpfulness and harmlessness of LLMs in enterprise settings. This will enable the early identification of malicious inputs while maintaining system efficiency, reliability and interpretability.

B. Problem Statement

Large Language Models are susceptible to prompt injection attacks, adversarial attacks, poisoning and privacy leakages. Current safeguards consist of primarily static or simple moderation that is unable to recognise context-based attacks.

C. Proposed Solution

Adaptive Multi-Layered Context-Aware Prompt Security and Output Governance (CAMS) takes a mixed layered approach to secure Large Language Models' interactions (LLMs). It adopt a pipeline architecture that believes in security for both inputs as well as outputs, the system seeks to mitigate prompt injection, attacks and information leakage. The system employs comprehensive prompt filtering, contextual adversarial filtering, and control over the outcomes. It starts with lexical and semantic analysis of the Prompts. To keep up with emerging threats, the system tracks the pattern history for recognise anomalies, semantic drift and manipulative tactics.

For multimodal information, the approach includes text, audio and image data. We convert audio inputs through speech-to-text methods and image inputs are fed through an Optical Character Recognition (OCR) process, so that the input into text for analysis.

On the output side, governance policies are used to identify sensitive data leak, unsafe responses, and policy violations. For code outputs, techniques such as sandboxing are keep program execution safe and secure. Further, abuse monitoring and user access control is applied. A monitoring dashboard and logs enable explainability and support auditability of system operation. In short, the proposed system is a modular pipeline involving prompt analysis, context analysis, multimodal pre-processing, and output validation and monitoring. This design enables secure, scalable and reliable LLM-based systems.

II. Literature Survey and Related Works

The research in “A Novel Evaluation Framework to Analyze Resiliency to Prompt Injection Attacks in Large Language Models” [1] proposes a series of evaluation metrics to assess the resilience of LLMs to prompt injection attacks. It allows for the comparison of model robustness but solely from an evaluation perspective, without offering a practical solution.

Likewise, "A Novel Framework to Mitigate Security Threats from Prompt Injection Attacks" [2] uses a combination of rule and semantic filtering to identify harmful prompts. It can detect many of these, but is static and thus lacks flexibility in handling the ever-evolving threat landscape.

“Defending Against Prompt Injection Attacks Using Automatic System Prompt Engineering" [3] research uses automatic prompt engineering of the system prompts to generate a dynamic defense that enhances model alignment. But this technique has been shown to be ineffective against sophisticated multi-step attacks.

Similarly, "Enhancing System Security: LLM-Driven Defense against Prompt Injection Vulnerabilities" [4] uses an AI-based security approach, which offers adaptive security. A key weakness of this approach is the fact that it is dependent on the functioning of the detector model.

For multimodal threats, “From Pixels to Prompts: A Systematic Study and Introduction to Image Prompt Injection Attacks” [5] shows how to use an image to embed an adversarial prompt for vision-language models. While highlighting these attacks, the research does not provide explicit protection against image-based attacks.

"Prompt Injection Detection and Mitigation with AI Multiagent NLP Based Agentic Frameworks" [6] provides a collaborative approach to threat detection, increasing detection rates through the use of multiagents. But it has significant computational and operational costs.

From the "attack" side, "Reconstruction-Based Prompt Generation Algorithm for Prompt Injection Attacks" [7] provides an effective way of generating adversarial prompts for prompt injection attacks. This study points out how safety features are easily bypassed, but does not suggest any defensive measures.

And finally, "A Survey of Attacks on Large Vision-Language Models: Resources, Advances, and Future Trends" [8] offers a taxonomy for classifying multimodal attacks but does not build a defense system.

For privacy protection, "Autonomous GDPR Compliance in Generative AI: Eliminating PII in LLM-Inspired Data Pipelines" [9] leverages monitoring and serverless orchestration to achieve PII redaction and GDPR compliance. This approach is effective at privacy protection, but makes the system very complicated.

For personalization, "Context-Aware Product Recommendations for Emerging Markets: A Sentiment-Driven LLM Approach" [10] leverages sentiment and context for more personable recommendations, but overlooks security concerns associated with these interactions.

Additional multimodal research includes “Image-Based Prompt Injection: Hijacking Multimodal LLMs through Visually Embedded Adversarial Instructions” [11], which demonstrates visual adversarial attacks can take over the system, with minimal defenses.

“Prompt Engineering Adversarial Attack against Image Captioning Models” [12] and "Unified Prompt Attack Against Text-to-Image Generation Models" [13] both prove successful adversarial attacks on vision products, though the latter uses multiple queries to fine-tune its gradient-based attack.

More subtle approaches are considered in "A Synergistic Multi-Agent Framework for Camouflage Attack on Large Language Models" [14], which employs sophisticated coordination between agents to craft stealthy attacks that are hard to counter without higher-level coordination.

Work on audio vulnerabilities in these models, such as “AUDIOJAILBREAK: Jailbreak Attacks against End-to-End Large Audio Language Models" [15] and "Audio Jailbreak Attacks: Exposing Vulnerabilities in SpeechGPT in a White-

Box Framework" [16], find audio interfaces open up many new attack surfaces that cannot be accounted for by traditional text-based models.

Previous advances like "A Conditional Context-Awareness Ontology-Based Personalized Recommendation Approach for E-Reading" [17] relied on ontology-based frameworks to enhance personalization, but has limited scalability for large environments.

More recently, "Automated Judging of LLM-Based Smart Contract Security Auditors" [18] uses an "LLM-as-judge" framework for AI auditors, which will be limited by the quality and potential biases of the model used.

Private-by-design systems such as “PrivMLLM: Efficient Three-Party Multimodal Large Language Model Secure Inference Supported Prompt Privacy” [19] implement secure multi-party computation for private inference, but have high overhead.

Surveys and reviews such as “Prompt-Based Jail breaking of Leading LLM Chatbots: A Survey of Attacks and Defenses” [20] describe a jailbreak taxonomy but lack a focus on implementation.

For specific cybersecurity applications, "Cognitive Security for AI-Native IIoT: An LLM-Centric Adaptive Defense Framework" [21] integrates LLMs and RAG but needs more practical testing.

“FireAI: A Framework for Dynamic Firewall Policy Generation using Generative Artificial Intelligence" [22] can automate firewall policies but risks generating automated rule errors.

Finally, “Privacy and Security Challenges in Large Language Models" [23] highlights risks and potential solutions, but lacks an integrated security solution.

III. RESEARCH GAP ANALYSIS

The existing security approaches lack "Context Understanding" by only looking at a single turn of the dialogue. This limited perspective cannot capture "slow-burn" or "salami-slicing" attacks, where a threat actor progressively drives the model to adopt a different persona or bypass a safety guardrail over multiple turns. The majority of filters lack state, so they do not track the dialogue over time to detect semantic drift or behavioral changes. Moreover, the best security often does require "Judge LLMs", this has a substantial resource requirement which leads to high costs to run on GPUs and much latency that renders them unsuitable for high-throughput and low-latency enterprise applications.

An additional important aspect is the existence of "Binary Blocking" and "Cross-Modal Vulnerabilities." Current deployments typically hard-reject entire prompts if even a single risky word is present, which not only annoys users, but also ruins the usability of non-offensive parts of the user requests. At the same time, existing solutions are mostly text-focused, leaving many doctrinal clues lurking in non-textual modes such as audio frequencies or other kinds of noise that exist in natural images & videos. There's a need for a multi-dimensional approach which seamlessly integrates lightweight, fast ML layers with intent generation pipelines to sanitize and translate multimodal inputs into safe intents without impinging either the UX or the system throughput.

IV. SYSTEM ARCHITECTURE

Our system architecture is a modular multi-layered framework bridging the user and the target LLM for enhanced security. Starting with a Multi-Modal Input Layer that accepts text, voice and video inputs using signal sanitization to block hidden adversarial frequencies or noise.

The sanitized input data is then further processed via a Text Normalization and Preprocessing Gateway, then forwarded to the Sliding Window Context Shift Analyzer (SWCSA). This layer keeps a buffer of prior prompts to compute a "Drift Score" derived from behavioral and entropy scores. When a prompt is detected, it passes through the Intent Fragmentation and Reconstruction (IFS-R) feature that fragments the prompt into sub-intents to identify the malicious component. And finally, Secure Output Validation, comparing against company policies ensures safe, reconstructed, instructions transmitted to the model but also an explainable audit trail through a central governance dashboard.

1. Input & Normalization Layer

The pipeline starts with the input of Multi-modal Inputs: text, audio (processed through Speech-to-Text) and images (processed through OCR Metadata extraction). Here, the processing system has a Signal Sanitization module that removes hidden embedded signals (such as steganography signals in the image and ultrasonic in audio). The sanitized data then proceeds to the Text Normalization component, which normalizes and tokenizes the input to provide a standardized representation to the rest of the layers.

2. Semantic & Conversational Analysis Layer

Normalized text is fed into a Sliding Window Context Buffer to track the most recent interactions and provide conversational context. The Sliding Window Context Shift Analyzer (SWCSA) tracks this to compute a Drift Score and measure entropy. This enables the system to recognize "slow-burn" attacks that gradually accumulate malicious intent over several interactions. In parallel, a Semantic & Lexical Filter examines keywords and structure to detect several types of attack in real-time.

3. Mitigation & Transformation Layer

If there is a risk or the prompt drift is high, it is directed to the Intent Fragmentation & Reconstruction (IFS-R) component. It disassembles the prompt into sub-intents to remove malicious content. It performs Secure Reconstruction, removing the sub-intents and isolating the sub-intents that were not malicious. This avoids "binary blocking" (completely blocking the user input) and improves the end-user experience.

4. Governance & Output Layer

The reconstructed and sanitized user input is then checked against pre-defined Security Policies and Finally Feed into the Target LLM. The resulting output then goes through an Output Guardrail to prevent leakage or malicious outputs. The lifecycle of each generation is tracked in the Governance & Explainability Dashboard, which logs the security controls and the audit trail of system administrators' reviews of the system's healing actions.

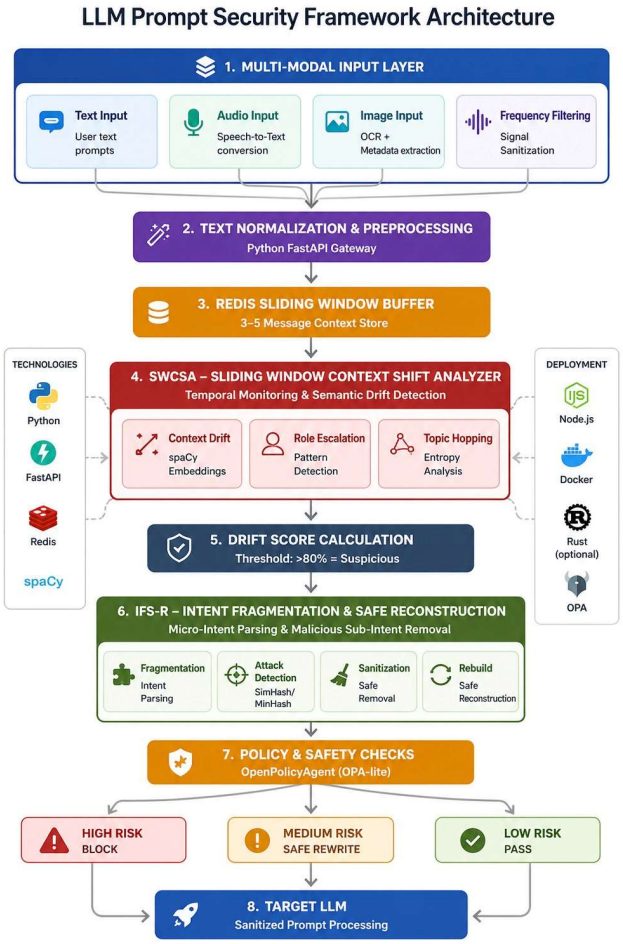


Fig. 1. System Architecture model

V. PROPOSED METHODOLOGY

The approach ensures a structured "Sanitize-Analyze-Mitigate" process to secure Large Language Model interactions with users. It comprises several phases:

Stage 1: Multi-Modal Ingestion and Sanitization

The process starts with Data Collection, where we collect adversarial patterns and legitimate input prompts for the detection model. Data Preprocessing sanitizes text, transcribes audio, extracts opto-character recognition (OCR) and metadata from images. For multi-modal security, Multi-Modal Input Processing performs signal sanitization, including audio and image denoising, to remove adversaries' content from non-text data.

Stage 2: Conversational Analysis

Filtered inputs enter Prompt Inspection and Filtering processes, in which lexical, semantic analysis, and known attack signatures are detected. To detect "slow-burn" attacks, the Sliding Window Context Shift Analyzer (SWCSA) examines past turns of the dialogue. This performs Drift Score Calculation; if the semantic drift, or role escalation, exceeds a threshold the chat is deemed a serious security risk.

Stage 3: Dynamic Mitigation and Intent Reconstruction

When a threat is identified, it uses Intent Fragmentation and Reconstruction (IFS-R). Rather than outright blocking, it breaks the user's request into sub-intents to block dangerous elements. This involves removing or rewriting segments of the prompt deemed harmful, and then putting it back together to achieve the safe intent of the user. The next step is Policy Control through a rule-based engine that categorises the risk level and allows final security checks.

Stage 4: Governance, Monitoring, and Assessment

The last stage is Post-Generation Governance where LLM-generated text is monitored for privacy leaks or malicious code, using sandbox environments. The Monitoring and Explainability component records system decisions, and offers clear reasons for all corrections to a dashboard. Lastly, the security framework is evaluated by conducting Execution and Validation through Evaluation Metrics including detection rate, false positives, and latency.

VI. TOOLS AND TECHNOLOGIES USED

The implementation uses a high-performance technology stack with Python FastAPI and Node.js as the main orchestration engine, and Rust micro-services for low-latency and high-speed processing of textual data. Redis is used for context management, with a sliding buffer for 3-5 messages. The NLP processing uses spaCy and Hugging Face Tokenizers on the CPU for semantic understanding. A security framework is integrated with the SWCSA algorithm for drift detection and IFS-R for intent graph matching. The framework is containerised with Docker and Kubernetes to facilitate deployment in a variety of environments, with a goal of less than 5ms latency and requiring at least 8GB of memory.

VII. EXPECTED OUTCOMES

This framework will likely lead to the identification of a large number of sophisticated, multi-turn prompt injection attacks that are not currently blocked by static filters. By replacing blacklist methods, the system will improve the user experience, with the Intent Fragmentation and Reconstruction (IFS-R) enabling the system to respond to benign user queries, despite their uncertainties. In addition, with the addition of a signal-to-intent pipeline, the "blind spots" of multimodal models will be closed, effectively neutralising hidden audio and visual threats. Finally, the

project will provide a resource-efficient, fast security gateway with enterprise governance, real-time explainability and a powerful defense mechanism capable of adapting to the ever-changing threat landscape.

VIII. Conclusion

This research project describes an Adaptive Context-Aware Multi-Layer Prompt Security and Output Governance Framework to tackle security vulnerabilities in Large Language Models (LLMs) such as prompt injection, manipulation, and leakage. The integration of multi-layer prompt inspection, context-aware analysis, multi-modal input processing and output governance makes this approach an integrated security solution, rather than a filter-based solution.

The framework improves the detection of direct and indirect attacks through context and pattern analysis. It also ensures user experience by generating safe prompts rather than input filtering. The use of text, audio, and image processing techniques guarantees consistent security across various input types, and the sandbox and policy enforcement techniques improve the security of the model's output. 24 The framework is also lightweight, scalable, and explainable, and can be readily deployed in real-world and enterprise environments.

The project not only showcases a real-world application of secure interactions with LLM, but also lays the groundwork for future research in AI safety and governance.

References

[1] G. Chen *et al.*, "AUDIOJAILBREAK: Jailbreak Attacks against End-to-End Large Audio-Language Models," *IEEE Transactions on Dependable and Secure Computing*, 2026, doi: 10.1109/TDSC.2026.3661073.

[2] B. Knowlton, J. Campa, D. S. Gallo, K. Dajani, and N. Alzahrani, "Prompt-Based Jailbreaking of Leading LLM Chatbots: A Survey of Attacks and Defenses," *IEEE Transactions on Artificial Intelligence*, 2026, doi: 10.1109/TAI.2026.3665656.

[3] X. Li, Z. Zhou, Z. Liu, Y. Wu, and W. Luo, "A Synergistic Multi-Agent Framework for Camouflage Attack on Large Language Models," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 34, 2026.

[4] J. Mao *et al.*, "Cognitive Security for AI-Native IIoT: An LLM-Centric Adaptive Defense Framework," in *Proc. Int. Conf. Computing, Networking and Communications (ICNC)*, 2026.

[5] V. K. Mavani, "An Autonomous Governance Framework for Generative AI: Real-Time PII Redaction and Compliance in LLM-Driven Data Pipelines," in *Proc. 5th IEEE Int. Conf. AI in Cybersecurity (ICAIC)*, Houston, TX, USA, Feb. 18-20, 2026.

[6] A. Yeo and D. Choi, "Multimodal Prompt Injection Attacks: Risks and Defenses for Modern LLMs," arXiv preprint arXiv:2509.05883v1, Sep. 2025.

[7] D. Liu, M. Yang, X. Qu, P. Zhou, Y. Cheng, and W. Hu, "A Survey of Attacks on Large Vision-Language Models: Resources, Advances, and Future Trends," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 11, Nov. 2025.

[8] D. Peng, Q. Ke, M. H. Huang, P. Hu, and J. Liu, "Unified Prompt Attack Against Text-to-Image Generation Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 6, June 2025.

[9] V. Rathod, S. Nabavirazavi, S. Zad, and S. S. Iyengar, "Privacy and Security Challenges in Large Language Models," in *Proc. IEEE 15th Annu. Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, Jan. 6-8, 2025.

[10] Y. Xue, L. Liu, Y. Luo, B. Sun, and S. Fu, "PrivMLLM: Efficient Three-Party Multimodal Large Language Model Secure Inference Supported Prompt Privacy," in *Proc. IEEE 33rd Int. Conf. Network Protocols (ICNP)*, 2025.

[11] V. Mothukuri and R. M. Parizi, "Automated Judging of LLM-based Smart Contract Security Auditors," in *Proc. IEEE Int. Conf. Blockchain and Cryptocurrency (ICBC)*, 2025.

[12] B. Ma, Z. J. Luo, H. Guo, and R. Duan, "Audio Jailbreak Attacks: Exposing Vulnerabilities in SpeechGPT in a White-Box Framework," in *Proc. 55th Annu. IEEE/IFIP Int. Conf. Dependable Systems and Networks Workshops (DSN-W)*, 2025.

[13] A. Rahmani, A. Abbas, and S. Sahki, "FireAI: A Framework for Dynamic Firewall Policy Generation using Generative Artificial Intelligence," in *Proc. 4th Int. Conf. Theoretical and Applicative Aspects of Computer Science (ICTAACS)*, 2025.

[14] N. Nagaraja, Z. Wang, B. Zhang, L. Zhang, and P. Patil, "Image-based Prompt Injection: Hijacking Multimodal LLMs through Visually Embedded Adversarial Instructions," in *Proc. 3rd Int. Conf. Foundation and Large Language Models (FLLM)*, 2025.

[15] A. Ahuja, "Context-Aware Product Recommendations for Emerging Markets: A Sentiment-Driven LLM Approach," in *Proc. 3rd Int. Conf. Business Analytics for Technology and Security (ICBATS)*, 2025.

[16] H. Vo, S. Yu, and X. Zheng, "Prompt Engineering Adversarial Attack against Image Captioning Models," in *Proc. 17th Int. Conf. Security of Information and Networks (SIN)*, 2024.

[17] Q. Gao, "A Conditional Context-Awareness Ontology based Personalized Recommendation Approach for E-Reading," in *Proc. 1st IEEE Int. Conf. Robotic Computing*, 2017.