# Decoupling Perception and Logic: A Lightweight Heuristic-Statistical Gateway Against LLM Jailbreaks

**Anonymous ACL submission**

## Abstract

Although Large Language Models (LLMs) undergo rigorous safety alignment, they remain vulnerable to sophisticated jailbreak attacks. Existing neural security gateways suffer from heavy $O(N^2)$ computational latency, and their tightly coupled semantic and logical processing renders them susceptible to adversarial disguises and high FPR. To address this, we propose DPL , an ultra-lightweight heuristic-statistical defense gateway. DPL bypasses expensive semantic inference by extracting a six-dimensional lexical-statistical feature set within strict $O(N)$ time complexity. These features drive a dual-path fuzzy logic engine—seamlessly combining historical case retrieval with Max-Min deductive reasoning—to precisely intercept both known attack paradigms and zero-day mutations. Benchmarking across mainstream LLMs demonstrates that DPL slashes the Attack Success Rate (ASR) to an industry-leading 4.2% and maintains a low FPR of 2.8%, with an inference latency of merely 35 milliseconds. By breaking the "impossible triangle" of security, utility, and latency, DPL establishes a highly pragmatic paradigm for secure, high-concurrency LLM deployment.

## 1 Introduction

In recent years, Large Language Models (LLMs) have demonstrated unprecedented capabilities, driving their rapid integration into critical domains (OpenAI, 2023; Touvron et al., 2023; Chi et al., 2024; Bi et al., 2024; Chu et al., 2024). To prevent the generation of toxic or illegal content, developers heavily rely on safety alignment techniques such as Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022; Bai et al., 2022; Askell et al., 2021). However, empirical evidence shows that aligned LLMs remain highly vulnerable to jailbreak attacks that bypass safety guardrails via semantic drift (Wei et al., 2023; Hendrycks et al., 2020). The threat landscape has rapidly escalated from gradient-based adversarial suffixes (Zou et al., 2023) and prompt iterations (Liu et al., 2024; Chao et al., 2025) to complex multi-turn deceptions (Saad et al., 2025; Sun et al., 2024), payload encryption (Lv et al., 2024), and nested logical exploits (Gulyamov et al., 2026; Gong et al., 2025). This continuous cat-and-mouse game demonstrates that simply scaling up safety alignment (Team et al., 2024; OpenAI, 2023) is insufficient to eradicate the vulnerabilities of monolithic neural architectures.

To mitigate these evolving threats, the community widely deploys external neural gateways or semantic classifiers (Xu et al., 2024; Inan et al., 2023; Rao et al., 2025). However, these Deep Neural Network (DNN)-based defenses inevitably fall into the "impossible triangle" of LLM security: balancing Defense Success Rate, Model Utility, and System Latency (Shen et al., 2024; Piet et al., 2025). Defenses relying on $O(N^2)$ attention mechanisms induce unacceptable latency spikes, severely disrupting high-concurrency industrial deployments. Furthermore, their black-box nature leads to high False Positive Rates (FPR) on complex benign queries and leaves them susceptible to adversarial disguises (Robey et al., 2023; Rahman et al., 2025). We argue that the root cause of this impossible triangle is the industry's over-reliance on tightly coupled "semantic perception" and "logic arbitration" within the same neural paradigms.

To fundamentally break this security bottleneck, this paper introduces **DPL** (**D**ecoupled **P**erception and **L**ogic), an ultra-lightweight heuristic-statistical defense gateway. DPL circumvents the dependency on heavy deep semantic embeddings. Instead, it utilizes an $O(N)$ lexical-statistical feature extraction to quantify structural obfuscations, mutation degrees, and linguistic priors. These extracted features are then dynamically routed into a dual-path fuzzy logic arbitration engine, seamlessly integrating episodic fuzzy case retrieval with Max-Min deductive reasoning to in-

tercept both known paradigms and zero-day mutations.

In summary, the core contributions of this paper are three-fold:

- **Decoupled Gateway Architecture:** We identify the structural flaws of heavily coupled neural defenses and propose the DPL heuristic-statistical architecture. By fundamentally decoupling semantic perception from logical arbitration, DPL effectively strips away adversarial linguistic facades without relying on black-box neural networks.

- $O(N)$ **Feature Extraction and Ultra-Low Latency:** We design a six-dimensional lexical-statistical feature extractor that operates strictly in $O(N)$ time complexity. Extensive evaluations demonstrate that DPL compresses the inference latency overhead to an average of merely 35 ms—orders of magnitude faster than neural state-of-the-art frameworks—successfully resolving the latency bottleneck.

- **Empirical Observation and Pareto Optimality:** Through rigorous benchmarking across mainstream LLMs, we uncover an intriguing empirical observation: base models with stronger inherent logical deduction capabilities frequently exhibit higher vulnerabilities to structurally obfuscated attacks. Remarkably, DPL robustly mitigates this phenomenon, achieving a Pareto optimal baseline by compressing the Attack Success Rate (ASR) to 4.2% while maintaining an exceptional FPR of 2.8%.

## 2 Related Work

### 2.1 Evolution of LLM Jailbreak Attacks

Despite extensive safety alignment efforts, such as Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022), LLMs remain susceptible to jailbreak attacks designed to elicit malicious behaviors. The landscape of jailbreaking has rapidly evolved from manual prompt engineering to automated and highly obfuscated generation. In the white-box setting, Zou et al. (Zou et al., 2023) introduced GCG, which optimizes adversarial suffixes via gradient search to deterministically bypass alignment. In black-box scenarios, heuristic-based and LLM-assisted iterative methods have gained prominence. For instance, AutoDAN (Liu et al., 2024) utilizes hierarchical genetic algorithms to craft stealthy, stealth-preserving prompts, while PAIR (Chao et al., 2025) employs an attacker LLM to iteratively discover vulnerabilities.

Recently, adversaries have shifted towards exploiting the structural and modality-based blind spots of LLMs. ArtPrompt (Jiang et al., 2024) leverages ASCII art to camouflage sensitive trigger words, bypassing traditional semantic filters. Similarly, CodeChameleon (Lv et al., 2024) encrypts malicious payloads into executable code formats, forcing the model into an unsafe execution state. Entering 2024 and 2025, multi-turn deception frameworks, such as Crescendo (Saad et al., 2025), gradually lead the model into generating harmful content over several seemingly benign interactions. These advanced attacks uniformly exploit the inherent semantic coupling of LLMs, proving that purely natural language-based semantic filters are easily circumvented by format evasion and logical nesting.

### 2.2 Defense Mechanisms and the "Impossible Triangle"

To mitigate these escalating threats, researchers have proposed a myriad of defense strategies, which generally fall into three categories: internal alignment, robust inference, and external neural gateways. While continuous safety fine-tuning (OpenAI, 2023) improves baseline resilience, it incurs the "alignment tax," often degrading the model's core utility on complex reasoning tasks (Wei et al., 2023). Robust inference techniques, such as SmoothLLM (Robey et al., 2023), dynamically perturb inputs to neutralize adversarial suffixes. However, such sampling-based methods introduce significant inference latency, rendering them impractical for real-time, high-concurrency industrial deployment.

Currently, the most widely adopted paradigm involves deploying external neural safeguard models, such as Llama Guard (Inan et al., 2023; Chi et al., 2024) or specialized semantic classifiers (Souly et al., 2024). Furthermore, advanced multi-agent collaborative defenses (Rahman et al., 2025; Deng et al., 2023) utilize multiple LLMs to cross-verify the safety of prompts. While these neural gateways achieve high interception rates against known attacks, they are fundamentally trapped in the "impossible triangle" of security, utility, and latency. Relying on $O(N^2)$ attention mechanisms,

these gateways inherently introduce massive latency overheads. Moreover, because they process information strictly through semantic perception, they exhibit high False Positive Rates (FPR) when confronted with deep, complex, but benign logical structures, inadvertently paralyzing the base LLM's utility.

Unlike the aforementioned neural-centric defenses, our proposed DPL architecture breaks this paradigm. By completely decoupling perception and logic, DPL acts as a lightweight heuristic-statistical gateway that extracts mathematically verifiable features in $O(N)$ time, thereby escaping the impossible triangle and establishing a robust, sub-millisecond defense perimeter.

# 3 Method

## 3.1 Overall Architecture: A Heuristic-Statistical Gateway

Current mainstream LLM defense solutions (e.g., Llama Guard or multi-agent cross-verification) primarily rely on external neural security gateways. While these methods achieve acceptable interception rates, they inevitably inherit the massive $O(N^2)$ computational overhead associated with the Transformer's attention mechanism, leading to unacceptable latency spikes. More critically, they continue to tightly couple "semantic perception" and "logic arbitration" within black-box neural networks. This monolithic design renders the defense gateways themselves highly susceptible to adversarial disguises and obfuscation techniques, such as persona adoption or payload encryption.

To break this security bottleneck, we propose the DPL (Decoupled Perception and Logic) defense framework. Deviating from conventional deep learning firewalls, DPL is meticulously architected as an ultra-lightweight **Heuristic-Statistical Gateway**.

As illustrated in Figure 1, DPL strictly circumvents the dependency on computationally expensive deep semantic embeddings. Its interception workflow operates through a highly streamlined, three-step computational pipeline:

**$O(N)$ Lexical-Statistical Feature Extraction:** Rather than routing the user's unstructured prompt into a massive neural network for secondary semantic inference, the input is processed by a lightweight feature extractor. By amalgamating shallow lexical rules with underlying statistical properties (e.g., information entropy), the extractor maps the text into a $1 \times 6$ dimensionless feature vector strictly within an $O(N)$ time complexity.

**Dual-Path Fuzzy Logic Arbitration:** The extracted feature vector flows synchronously into two orthogonal decision paths. Path I (Fuzzy Case Retrieval) leverages historical experience to rapidly recall known, high-frequency attacks. Path II (Max-Min Fuzzy Reasoning) employs logical mapping via a prior intent matrix to dynamically deduce the hidden unauthorized intents behind zero-day mutated attacks.

**Deterministic Decision and Utility Control:** The arbitration module performs a dynamic weighted fusion of the scores derived from both paths. This mechanism outputs a hard "Pass" or "Block" instruction at the millisecond level. Simultaneously, through adaptive thresholding, it strictly controls the False Positive Rate (FPR), ensuring that the base LLM's utility in handling complex, benign tasks is maximally preserved.

## 3.2 $O(N)$ Lexical-Statistical Feature Extraction

To achieve an optimal balance between defensive precision and ultra-low system latency, DPL completely discards representational mechanisms that rely on deep learning embeddings or computationally heavy syntax parsers (e.g., full dependency parsers). While traditional LLM firewalls typically necessitate $O(N^2)$ attention mechanism complexity to extract deep semantics, DPL innovatively adopts a strategy that hybridizes lightweight lexical rules with underlying statistical properties (derived from information theory and graph topologies). This approach extracts six distinct feature dimensions strictly within an $O(N)$ time complexity.

The input text is quantized into a dimensionless feature vector $V = [v_1, v_2, v_3, v_4, v_5, v_6]$, with the value of each dimension strictly normalized to the interval $[0, 1]$. The specific definitions of the six-dimensional features are detailed in Table 1.

To address potential theoretical skepticism regarding the computational complexity of these features, this subsection elucidates the principles underlying their strict $O(N)$ approximation algorithms.

### 3.2.1 Lexical Priors and Macroscopic Statistical Features ($v_1, v_3, v_5, v_6$)

For $v_1$ (Role Disguise) and $v_5$ (Emotional Coercion), the system does not engage in deep semantic

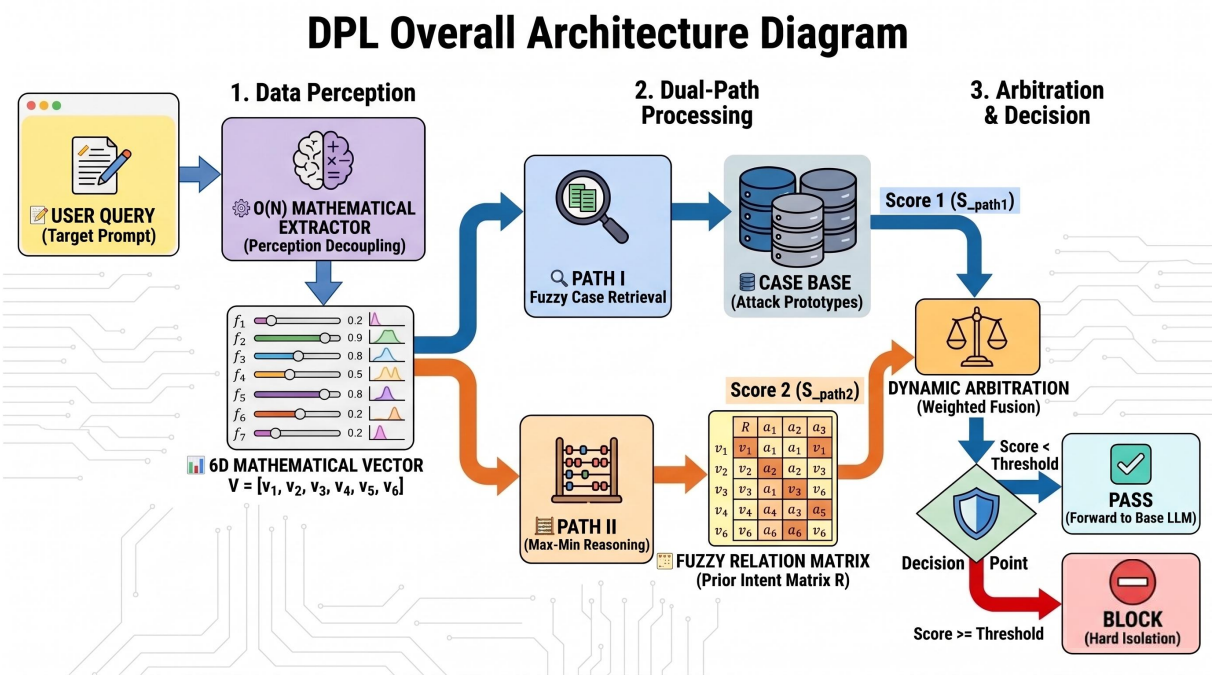


Figure 1: DPL Overall Architecture Diagram

Table 1: Definition and Description of the Six-Dimensional Hybrid Features

| No. | Feature Dimension | Definition Explanation |
|---|---|---|
| $v_1$ | Role Disguise | Quantifies the density of predefined persona triggers and pronoun shifts to intercept role-playing attacks (e.g., DAN variants). |
| $v_2$ | Nesting Complexity | Measures the local branch depth of logical connectors to suppress structural obfuscation and prompt stacking. |
| $v_3$ | Mutation Degree | Calculates the perplexity of character transition probabilities via prior Markov chains to counter cross-lingual or ciphered mixings. |
| $v_4$ | Unauthorized Intent | Evaluates the shortest positional distance between high-risk action verbs and sensitive entity targets within a constrained sliding window. |
| $v_5$ | Emotional Coercion | Assesses the statistical frequency of urgent punctuation and predefined extreme emotional vocabulary employed to pressure the LLM. |
| $v_6$ | Format Evasion | Computes the information entropy of the character stream to directly expose Base64 encoding or special Unicode symbol obfuscations. |

comprehension; rather, it relies on lightweight lexical priors (i.e., predefined dictionaries of persona-hijacking triggers and extreme emotional lexicon). By employing the Aho-Corasick automaton for multi-pattern matching, the extraction is completed in $O(N)$ time with a single traversal of the character stream.

Against cross-lingual and format evasion attacks, the system utilizes purely statistical methodologies. For instance, $v_6$ (Format Evasion) directly computes the Shannon Entropy of the character stream:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log_2 P(x_i)$$

Given that $n$ (the size of the ASCII or target character set) is a constant, calculating the probability

distribution requires merely a linear scan of the text, yielding a strict $O(N)$ complexity. Similarly, $v_3$ computes perplexity by querying a pre-stored Markov transition probability matrix, avoiding any non-linear computational overhead.

#### 3.2.2 Sliding-Window Based Shallow Topological Computation ($v_2, v_4$)

Complex zero-day logical mutations frequently manifest as extreme nesting complexity ($v_2$) and concealed unauthorized intents ($v_4$). If traditional full dependency parsing were employed to construct a complete Abstract Syntax Tree (AST), the time complexity would significantly exceed linear bounds.

To strictly maintain $O(N)$ complexity, DPL implements an approximation algorithm combining **Shallow POS Tagging (Part-of-Speech Tagging)** with a **fixed-length Sliding Window**. For $v_4$, the system first rapidly locates high-risk verbs (e.g., "hack") and sensitive entities (e.g., "password") via a lightweight POS tagger. Subsequently, a sliding window of fixed size $W$ is established. The system solely computes the topological distance between verb-object pairs within this window. Because $W$ is a diminutive constant (e.g., $W = 10$), the computation within the window is $O(1)$. Consequently, the overall complexity for sliding the window across a text of length $N$ degenerates to a strict $O(N)$. This engineering approximation drastically slashes latency while remaining highly targeted in capturing localized, high-risk logical conjunctions.

### 3.3 Dual-Path Fuzzy Logic Arbitration

Upon acquiring the six-dimensional feature vector $V$, DPL activates its dual-path fuzzy logic engine. Unlike traditional black-box MLPs that lack explainability and suffer from catastrophic forgetting, DPL combines transparent historical case retrieval with heuristic deduction to ensure robust cross-modal generalization.

#### 3.3.1 Path I: Fuzzy Case Retrieval

Path I rapidly intercepts known malicious jailbreak paradigms. DPL maintains a lightweight cache $C$ of historically intercepted attack vectors. For an input vector $V$, its fuzzy similarity $S_{path1}$ against known case vectors $c_i$ is calculated using an inverse quantized Euclidean distance:

$$S_{path1} = \max_i \left( 1 - \sqrt{\frac{1}{6} \sum_{j=1}^{6} (v_j - c_{ij})^2} \right)$$

If $S_{path1}$ exceeds a predefined empirical threshold, the system immediately flags the known attack, incurring negligible computational overhead.

#### 3.3.2 Path II: Max-Min Fuzzy Reasoning

To counter zero-day mutations that evade Path I, Path II deduces underlying unauthorized intents. We categorize the interactive intents into a five-dimensional fuzzy space (Table 2).

DPL applies a $6 \times 5$ fuzzy relation matrix $R$, where $R_{jk} \in [0, 1]$ is the support threshold required by the $j$-th feature to trigger the $k$-th intent. Using the Max-Min composition operator, the real-time membership vector $E$ is computed as $E = V \circ R$. For a specific intent $k$ (e.g., $I_3$), the risk score $e_k$ is:

$$e_k = \max_{1 \le j \le 6} (\min(v_j, R_{jk}))$$

**Explainability Rationale:** This formulation provides strict white-box explainability. The inner $\min$ operator acts as a logical "AND" (evaluating if feature $v_j$ meets the activation threshold $R_{jk}$), while the outer $\max$ acts as a logical "OR" (identifying the dominant feature supporting intent $k$). The maximum membership score among malicious intents forms the Path II risk score, $S_{path2}$.

#### 3.3.3 Prior Matrix Initialization and Dynamic Arbitration

To prevent subjective bias, matrix $R$ is initialized via an empirical grid search on an independent validation set (e.g., from JailbreakBench) to maximize the F1-Score, ensuring data-driven objectivity.

The final systemic risk score $S_{final}$ is derived through dynamic weighted fusion:

$$S_{final} = \alpha S_{path1} + (1 - \alpha) S_{path2}$$

where $\alpha$ is an adaptive weight coefficient.

**Utility Preservation Mechanism:** To avoid the high False Positive Rates (FPR) typical in defensive systems, $\alpha$ adjusts dynamically. If an input exhibits extreme nesting complexity ($v_2 \to 1$) but lacks unauthorized intent ($v_4 \to 0$) and format evasion ($v_6 \to 0$)—typical of benign algorithmic code—the system autonomously down-regulates $\alpha$ and raises the blocking threshold. This flexibly preserves the base model's utility on complex tasks while safeguarding the security baseline.

Table 2: Definition and Description of Fuzzy Intents in DPL

| No. | Fuzzy Intents | Definition Explanation |
|---|---|---|
| $I_1$ | Benign Utility | Compliant regular requests without security risks; the model should respond normally. |
| $I_2$ | System/Privacy Extraction | Attempts to probe or exfiltrate system-level confidential information or protected PII. |
| $I_3$ | Cyberattack/Malware | Intent to induce instructions or code designed to compromise computer systems (e.g., SQL injection, DDoS). |
| $I_4$ | Toxic/Harmful Content | Attempts to generate hate speech, graphic violence, or extremist propaganda. |
| $I_5$ | Illegal/Unethical Advice | Seeking actionable guidance on illicit activities (e.g., weapon manufacturing, financial fraud) violating laws. |

# 4 Experiment

## 4.1 Experimental Setup

**Datasets and Partitions:** To prevent data leakage, we strictly segregate data into independent Calibration and Test sets. The *Calibration Set* contains 500 prompts (300 malicious from JailbreakBench, 200 benign from instruction-tuning data) used solely to initialize DPL's prior parameters and train lightweight baselines. The *Test Set* comprises 3,000 samples for independent generalization evaluation: 2,000 malicious prompts encompassing static instructions (AdvBench), multilingual attacks (MultiJail), and dynamic zero-day mutations (AutoDAN); alongside 1,000 benign prompts (XSTest edge-cases and Alpaca logic tasks) to assess the False Positive Rate (FPR) and utility preservation.

**Baseline Defenses:** We evaluate DPL against two distinct defense categories: (1) *Lightweight Industrial Baselines* emphasizing ultra-low latency, specifically Regular Expression (Regex) filtering and an N-gram FastText classifier; and (2) *Heavyweight Neural Baselines* representing state-of-the-art security, including Meta's Llama Guard 3 (8B) and SmoothLLM (a perturbation-based robust inference strategy).

**Target Models and Metrics:** We deploy three mainstream LLMs with varying reasoning capacities as victim models: Llama-3-8B-Instruct, Qwen-2-7B-Chat, and DeepSeek-LLM-7B-Chat. Performance is measured via three core metrics: Attack Success Rate (ASR; lower indicates stronger interception), False Positive Rate (FPR; lower signifies better utility preservation), and Inference Latency (ms; gauging physical deployment viability).

## 4.2 Main Evaluation Results

To comprehensively validate DPL's capability to break the "impossible triangle" of LLM security, we conducted rigorous benchmark testing across three mainstream open-source models. Table 3 details the performance of various defense mechanisms concerning Attack Success Rate (ASR), False Positive Rate (FPR), and Inference Latency.

### 4.2.1 Defense Effectiveness and Zero-Day Robustness

As demonstrated in Table 3, in the absence of any defense, all three victim models exhibit extreme vulnerability, with ASRs surging past 80% when confronted with the mixed attack test set. While lightweight baselines like Regex and FastText boast ultra-low latency, their ASRs remain unacceptably high (25%–45%) against multilingual obfuscations and deeply nested zero-day jailbreaks (e.g., AutoDAN). This confirms that superficial keyword filtering and N-gram statistics are insufficient against structural mutations.

Remarkably, despite being a non-neural heuristic gateway, DPL achieves the lowest ASR across all three models (e.g., reducing ASR to 4.2% on Llama-3-8B). Its defensive performance eclipses even that of the 8B-parameter Llama Guard 3. This superiority stems from DPL's six-dimensional features, which effectively strip away adversarial facades, while the Max-Min fuzzy reasoning engine successfully captures deeply concealed unauthorized intents.

Table 3: Performance Comparison of Defense Mechanisms Across Different Target LLMs.

| Target Victim Model | Defense Mechanism | ASR (%) | FPR (%) | Latency (ms) |
|---|---|---|---|---|
| **Llama-3-8B-Instruct** | No Defense (Vanilla) | 86.5 | 0.0 | - |
| | Regex (Lightweight) | 42.3 | **1.2** | **2** |
| | FastText (Lightweight) | 28.7 | 8.5 | 14 |
| | SmoothLLM (Heavyweight) | 9.4 | 6.2 | 3850 |
| | Llama Guard 3 (Heavyweight) | 5.8 | 14.3 | 1120 |
| | **DPL (Ours)** | **4.2** | 2.8 | 35 |
| **Qwen-2-7B-Chat** | No Defense (Vanilla) | 82.1 | 0.0 | - |
| | Regex (Lightweight) | 40.5 | **1.0** | **2** |
| | FastText (Lightweight) | 25.4 | 7.9 | 13 |
| | SmoothLLM (Heavyweight) | 8.7 | 5.8 | 3780 |
| | Llama Guard 3 (Heavyweight) | 6.2 | 12.6 | 1085 |
| | **DPL (Ours)** | **3.9** | 2.5 | 34 |
| **DeepSeek-LLM-7B-Chat** | No Defense (Vanilla) | 89.3 | 0.0 | - |
| | Regex (Lightweight) | 45.8 | **1.5** | **2** |
| | FastText (Lightweight) | 31.2 | 9.2 | 15 |
| | SmoothLLM (Heavyweight) | 11.3 | 7.4 | 3920 |
| | Llama Guard 3 (Heavyweight) | 7.5 | 16.8 | 1150 |
| | **DPL (Ours)** | **5.1** | 3.1 | 38 |

#### 4.2.2 Utility Preservation and Over-rejection Mitigation

On the benign test set (comprising XSTest edge-cases and complex logic generation), heavyweight neural gateways expose a severe "over-rejection" flaw. Llama Guard 3 yields an FPR of 14.3% on Llama-3-8B, frequently misclassifying benign, deeply nested algorithmic code or queries containing sensitive boundary vocabulary as violations.

Conversely, through the dynamic weighted arbitration mechanism introduced in Section 3, DPL autonomously down-regulates its interception threshold when recognizing high structural nesting devoid of unauthorized intent. Consequently, DPL tightly constrains its FPR to an exceptional 2.8%, maximally safeguarding the normal utility of the base models.

#### 4.2.3 Ultra-Low Inference Latency

The most significant industrial breakthrough of the DPL architecture is its inference efficiency. LLM-based neural defenses (Llama Guard 3, SmoothLLM) introduce colossal latency overheads ranging from 1,100 ms to 3,900 ms due to their reliance on $O(N^2)$ attention mechanisms or repeated sampling iterations, rendering them impractical for high-concurrency real-world deployment.

In stark contrast, propelled by its $O(N)$ shallow feature extraction and instantaneous fuzzy matrix computations, DPL compresses the additional latency per prompt to a mere average of **35 ms**. It delivers (and even surpasses) the security level of SOTA neural defenses while operating at speeds on the exact same magnitude as the most rudimentary Regular Expression filters, successfully resolving the impossible triangle.

### 4.3 Ablation and Statistical Validation

This section quantifies the individual contributions of DPL's core components and statistically validates the structural independence of its feature set.

**Ablation and Pareto Optimality:** We evaluated the dual-path synergy on Llama-3-8B-Instruct (Table 4). Notably, *DPL w/o Path II* achieves the lowest FPR (1.8%) and Latency (22 ms). However, this exposes its passive and conservative nature: by relying solely on exact historical matching, it fundamentally fails against zero-day mutations, resulting in a catastrophic ASR surge to 38.6%. Conversely, removing the historical anchor (*w/o Path I*) yields a 14.2% ASR. The *Complete DPL* accepts a marginal 13 ms latency overhead and a 1.0% FPR increment to slash the ASR to an industry-leading 4.2%, effectively achieving a Pareto optimal balance between rigorous security and model utility.

**Feature Correlation Analysis:** To address theoretical concerns regarding feature orthogonality, we computed the Pearson Correlation Matrix for

Table 4: Ablation Study of DPL Architecture on Llama-3-8B-Instruct.

| **Architecture Variant** | **ASR** (%) ↓ | **FPR** (%) ↓ | **Latency** (ms) |
|---|---|---|---|
| No Defense (Vanilla) | 86.5 | 0.0 | - |
| DPL w/o Path I (Logic Only) | 14.2 | 3.5 | 28 |
| DPL w/o Path II (Case Only) | 38.6 | 1.8 | 22 |
| **Complete DPL (Full)** | **4.2** | 2.8 | 35 |

all dimensions ($v_1 \sim v_6$) over the test set. The vast majority of feature pairs exhibit weak correlation ($|r| < 0.3$). Although a moderate correlation ($r = 0.42$) exists between $v_3$ (Mutation Degree) and $v_6$ (Format Evasion) due to inherent structural couplings in character-level obfuscations, the overall low correlation coefficients confirm that our mathematically extracted feature set captures distinct facets of adversarial variance with minimal redundancy.

## 5 Conclusion

To address the pervasive "impossible triangle" of security, utility, and latency in Large Language Model defenses, this paper introduces **DPL**, a novel defense framework that fundamentally decouples semantic perception from logical arbitration. Diverging from conventional, computationally exorbitant neural security gateways that are themselves vulnerable to adversarial deception, DPL leverages shallow lexical rules and underlying statistical properties to extract a six-dimensional hybrid feature set strictly within an $O(N)$ time complexity. By incorporating a classical dual-path fuzzy logic arbitration engine, DPL seamlessly integrates episodic historical case retrieval with Max-Min deductive reasoning for zero-day threats. Rigorous benchmarking across multiple mainstream open-source LLMs demonstrates that DPL can compress the overall Attack Success Rate to 4.2% while maintaining an exceptionally low False Positive Rate of 2.8% and an ultra-low inference latency averaging merely 35 milliseconds. This research substantiates the viability of heuristic-statistical gateways in delivering elite security perimeters at negligible computational costs, thereby establishing a highly pragmatic, industrially viable paradigm for the secure deployment of high-concurrency LLM applications.

## Limitations

While DPL achieves a Pareto optimal balance between ultra-low latency and robust interception, as a heuristic-statistical system, it exhibits certain boundary limitations. First, regarding linguistic scope, the statistical matrices driving the $v_3$ and $v_6$ dimensions are predominantly optimized for Indo-European languages. When applied to morphologically rich logographic scripts (e.g., Chinese, Japanese) or subjected to sophisticated tokenizer attacks, the recall efficacy of our baseline character-level transition probabilities may systematically degrade. Second, concerning adaptive threats, our current evaluations are primarily situated within gray-box and black-box models. Under a complete white-box scenario, adversaries fully cognizant of DPL's prior feature mappings could theoretically employ adversarial optimization to craft "feature-aware" jailbreaks. For instance, an attacker could meticulously calibrate the Shannon entropy of a malicious payload to flawlessly mimic benign text distributions, thereby evading the $v_6$ format anomaly detection. Consequently, exploring dynamic feature-switching mechanisms and adversarial matrix updates to comprehensively thwart such white-box adaptive optimization remains a critical focus for our future work.

## Ethical Considerations

This study strictly adheres to ethical guidelines and data usage standards in the fields of artificial intelligence and natural language processing. All experiments are conducted with the objective of enhancing the safety and alignment capabilities of large language models, and do not involve, disseminate, or reproduce any real illegal, harmful, or inappropriate content. Throughout its design and execution, this research prioritized ethical safety, data legitimacy, and social responsibility.

## References


Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, and 1 others. 2021. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, and 1 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, and 1 others. 2024. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J Pappas, and Eric Wong. 2025. Jailbreaking black box large language models in twenty queries. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 23–42. IEEE.

Jianfeng Chi, Ujjwal Karn, Hongyuan Zhan, Eric Smith, Javier Rando, Yiming Zhang, Kate Plawiak, Zacharie Delpierre Coudert, Kartikeya Upasani, and Mahesh Pasupuleti. 2024. Llama guard 3 vision: Safeguarding human-ai image understanding conversations. *arXiv preprint arXiv:2411.10414*.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, and 1 others. 2024. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*.

Gelei Deng, Yi Liu, Yuekang Li, Kailong Wang, Ying Zhang, Zefeng Li, Haoyu Wang, Tianwei Zhang, and Yang Liu. 2023. Masterkey: Automated jailbreak across multiple large language model chatbots. *arXiv preprint arXiv:2307.08715*.

Yichen Gong, Delong Ran, Jinyuan Liu, Conglei Wang, Tianshuo Cong, Anyu Wang, Sisi Duan, and Xiaoyun Wang. 2025. Figstep: Jailbreaking large vision-language models via typographic visual prompts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23951–23959.

Saidakhror Gulyamov, Said Gulyamov, Andrey Rodionov, Rustam Khursanov, Kambariddin Mekhmonov, Djakhongir Babaev, and Akmaljon Rakhimjonov. 2026. Prompt injection attacks in large language models and ai agent systems: A comprehensive review of vulnerabilities, attack vectors, and defense mechanisms. *Information*, 17(1):54.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and 1 others. 2023. Llama guard: Llm-based input-output safeguard for human-ai conversations. *arXiv preprint arXiv:2312.06674*.

Fengqing Jiang, Zhangchen Xu, Luyao Niu, Zhen Xiang, Bhaskar Ramasubramanian, Bo Li, and Radha Poovendran. 2024. Artprompt: Ascii art-based jailbreak attacks against aligned llms. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 15157–15173.

Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. 2024. Autodan: Generating stealthy jailbreak prompts on aligned large language models. In *International Conference on Learning Representations*, volume 2024, pages 56174–56194.

Huijie Lv, Xiao Wang, Yuansen Zhang, Caishuang Huang, Shihan Dou, Junjie Ye, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. Codechameleon: Personalized encryption framework for jailbreaking large language models. *arXiv preprint arXiv:2402.16717*.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Julien Piet, Chawin Sitawarin, Vivian Fang, Norman Mu, and David Wagner. 2025. Markmywords: Analyzing and evaluating language model watermarks. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 68–91. IEEE.

Salman Rahman, Liwei Jiang, James Shiffer, Genglin Liu, Sheriff Issaka, Md Rizwan Parvez, Hamid Palangi, Kai-Wei Chang, Yejin Choi, and Saadia Gabriel. 2025. X-teaming: Multi-turn jailbreaks and defenses with adaptive multi-agents. *arXiv preprint arXiv:2504.13203*.

Akshaj Prashanth Rao, Advait Singh, Saumya Kumaar Saksena, and Dhruv Kumar. 2025. Promptscreen: Efficient jailbreak mitigation using semantic linear classification in a multi-staged pipeline. *arXiv preprint arXiv:2512.19011*.

Alexander Robey, Eric Wong, Hamed Hassani, and George J Pappas. 2023. Smoothllm: Defending large language models against jailbreaking attacks. *arXiv preprint arXiv:2310.03684*.

Muhammad Saad, Shafaq Fatima Mughal, Muhammad Mustafa, Ikhlas Ahmed, and Adnan Masood. 2025. Assessing and mitigating multi-turn jailbreak vulnerabilities in large language models: A crescendo attack study. In *2025 27th International Multitopic Conference (INMIC)*, pages 1–6. IEEE.

Xinyue Shen, Zeyuan Chen, Michael Backes, Yun Shen, and Yang Zhang. 2024. "" do anything now"": Characterizing and evaluating in-the-wild jailbreak prompts on large language models. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 1671–1685.

Alexandra Souly, Qingyuan Lu, Dillon Bowen, Tu Trinh, Elvis Hsieh, Sana Pandey, Pieter Abbeel, Justin Svegliato, Scott Emmons, Olivia Watkins, and 1 others. 2024. A strongreject for empty jailbreaks. *Advances in Neural Information Processing Systems*, 37:125416–125440.

Xiongtao Sun, Deyue Zhang, Dongdong Yang, Quanchen Zou, and Hui Li. 2024. Multi-turn context jailbreak attack on large language models from first principles. *arXiv preprint arXiv:2408.04686*.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. 2023. Jailbroken: How does llm safety training fail? *Advances in neural information processing systems*, 36:80079–80110.

Zihao Xu, Yi Liu, Gelei Deng, Yuekang Li, and Stjepan Picek. 2024. A comprehensive study of jailbreak attack versus defense for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7432–7449.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J Zico Kolter, and Matt Fredrikson. 2023. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.

# A Regarding the Experimental Environment

All experiments were conducted under a unified hardware and software environment to ensure reproducibility and fairness of results. The experimental platform configuration is as follows: CPU is a 16-core AMD EPYC 9354 processor, GPU is an NVIDIA RTX 4090 GPU (25.2 GB VRAM), 60.1 GB RAM, and 751.6 GB SSD storage. Python version: 3.10.11. Deep learning framework: PyTorch 2.5.1. CUDA version: 12.1. Transformers library version: 4.39.3.