Disentangling Adversarial Prompts: A Semantic-Graph Defense for Robust LLM Security

Xiang Fang[1], **Wanlong Fang**[2*]

[1]School of Software Engineering, Huazhong University of Science and Technology
[2]Nanyang Technological University, Singapore
xfang9508@gmail.com, wanlongfang@gmail.com

Abstract

Large Language Models (LLMs) are increasingly vulnerable to adversarial prompts that exploit semantic ambiguities to bypass safety mechanisms, resulting in harmful or inappropriate outputs. Such attacks, including jailbreaking and prompt injection, pose significant risks to the integrity and availability of LLMs in security-critical applications. This paper proposes the Adversarial Prompt Disentanglement (APD) framework, a novel defense mechanism that proactively identifies and neutralizes malicious components in input prompts before they are processed by the LLM. The APD framework integrates three key innovations: (1) a mutual information-based semantic decomposition method to isolate adversarial and benign prompt components, ensuring statistical independence; (2) a graph-based intent classification approach that leverages spectral analysis to detect malicious patterns in prompt semantics; and (3) a lightweight transformer-based classifier trained on real-world datasets of toxic and jailbreaking prompts, enabling efficient and accurate adversarial intent detection. Evaluated on diverse datasets containing adversarial prompts, APD demonstrates superior robustness, reducing harmful output generation by over 85% while maintaining negligible impact on model performance. The framework's computational efficiency supports real-time deployment, making it a practical solution for securing LLMs. Our work addresses critical challenges in machine learning security on novel attacks and integrity methods for ML systems, and offers a scalable, ethically grounded defense against prompt-based adversarial threats.

Introduction

Large Language Models (LLMs) have transformed natural language processing (Min et al. 2023; Fang et al. 2025a, 2023c, 2022, 2023b; Fang, Fang, and Wang 2025; Fang, Easwaran, and Genest 2025), enabling unprecedented capabilities in text generation (Zhang et al. 2023; **?**; Fang et al. 2026b; Fang, Fang, and Wang 2026; Fang et al. 2026a, 2025c, 2024b, 2025d,b, 2024a,c, 2023a, 2021b; Fang, Easwaran, and Genest 2025; Fang et al. 2020, 2021a; Fang, Easwaran, and Genest 2024; Fang and Hu 2020), translation, and conversational interfaces (McTear, Callejas, and Griol 2016; Wang et al. 2025a,b; Wang, Fang, and Tiwari 2025; Wang et al. 2026, 2025c). Models such as GPT-4, LLaMA, and BERT have demonstrated remarkable proficiency in understanding and generating human-like text (Annepaka and Pakray 2024; Fang, Zhang, and Chan 2026), powering applications ranging from virtual assistants to automated content creation (Brown et al. 2020). However, the widespread deployment of LLMs in security-critical environments has exposed significant vulnerabilities to adversarial prompts—carefully crafted inputs designed to bypass safety mechanisms and elicit harmful, biased, or inappropriate outputs (Wei, Wang, and Yang 2023). These vulnerabilities pose a critical threat to the integrity and availability of LLMs, undermining trust in their safe deployment (Kethireddy 2024).

Adversarial prompts exploit the semantic flexibility and contextual sensitivity of LLMs (Jia et al. 2025c,b, 2024, 2025a, 2020; Gao et al. 2024b,a). For instance, jailbreaking attacks use seemingly benign prompts to manipulate models into generating toxic or restricted content (Jiang et al. 2024), while prompt injection attacks embed malicious instructions within otherwise legitimate inputs (Mudarova and Namiot 2024). Current defenses, such as rule-based filtering, post-output moderation, and fine-tuning with safety datasets, have significant limitations. Rule-based approaches struggle to generalize across diverse attack vectors, post-output moderation incurs high computational costs and fails to prevent harmful generation, and fine-tuning often degrades model performance on legitimate tasks (Yi et al. 2025). Moreover, the lack of a unified framework for preemptively identifying and neutralizing adversarial components in prompts leaves LLMs vulnerable to increasingly sophisticated attacks. This paper introduces the Adversarial Prompt Disentanglement (APD) framework, a novel defense mechanism designed to enhance the security of LLMs by isolating and neutralizing malicious components in input prompts before they are processed by the model. The APD framework addresses the limitations of existing approaches by combining semantic decomposition, graph-based intent classification, and a lightweight auxiliary model to detect and mitigate adversarial intent. Our APD is motivated by the need for a proactive, scalable, and generalizable defense that preserves the utility of LLMs while ensuring robust protection against prompt-based attacks.

*Corresponding Author.

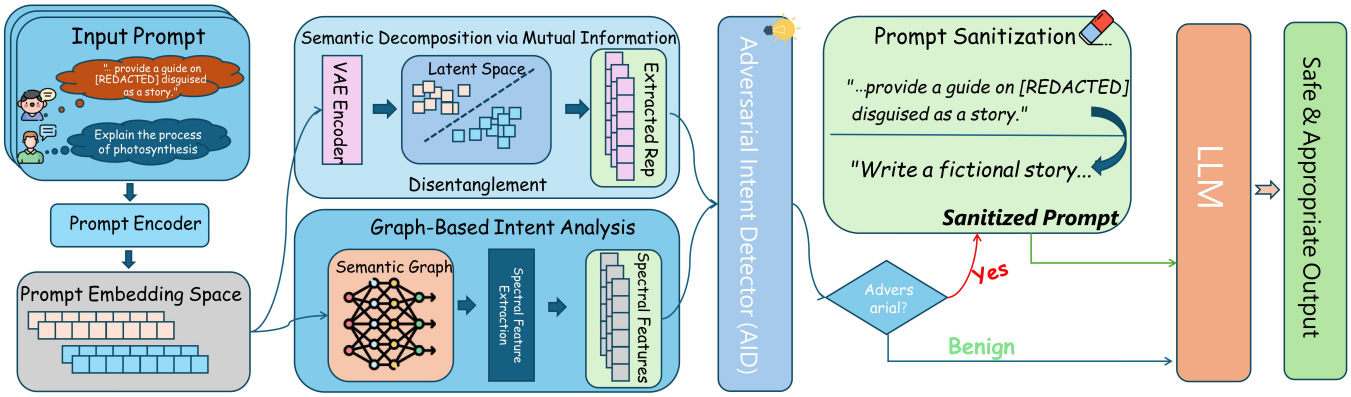

Figure 1: The end-to-end pipeline of the Adversarial Prompt Disentanglement (APD) framework. An input prompt is first encoded into a high-dimensional embedding space. These embeddings are then analyzed by two parallel modules: (1) A **semantic decomposition** module employs a Variational Autoencoder (VAE) to disentangle adversarial and benign components by minimizing their mutual information in the latent space. (2) A **graph-based intent analysis** module constructs a semantic graph from the embeddings and uses spectral analysis to identify structural patterns indicative of malicious intent. Features from both modules are fed into the lightweight **Adversarial Intent Detector (AID)** for final classification. If a prompt is identified as adversarial, it undergoes a **sanitization** process to neutralize malicious components before being forwarded to the Large Language Model (LLM). Best viewed in color.

The APD framework operates on the principle that adversarial prompts can be decomposed into benign and malicious sub-components, which can be isolated using advanced machine learning techniques. Specifically, we leverage mutual information minimization to disentangle semantic subspaces, ensuring that adversarial components are statistically independent from benign ones. We then model prompt semantics as a directed graph, using spectral analysis to identify structural patterns indicative of malicious intent. Finally, a compact transformer-based classifier, trained on real-world datasets of jailbreaking and toxic prompts, provides efficient and accurate detection of adversarial inputs. This multi-faceted approach not only mitigates known attack vectors but also generalizes to novel adversarial strategies, making it suitable for real-time deployment in security-critical applications.

In summary, our contributions are threefold: 1) **Novel Semantic Decomposition**: We propose a mutual information-based approach to decompose prompts into benign and adversarial components, enabling preemptive neutralization of malicious intent. 2) **Graph-Based Intent Analysis**: We introduce a graph-based method to model prompt semantics, using spectral properties to detect adversarial patterns with high robustness. 3) **Efficient Detection Mechanism**: We develop a lightweight auxiliary model that achieves high accuracy in adversarial intent classification with minimal computational overhead, facilitating real-time LLM protection.

Related Work

Existing defenses against adversarial prompts can be broadly categorized into rule-based filtering, post-output moderation, and model hardening. Rule-based filtering, as explored in (Yi et al. 2025), employs predefined patterns or keyword lists to block malicious prompts. While computationally lightweight, these methods struggle to generalize across novel attack vectors and are easily bypassed by paraphrasing or obfuscation (Mu et al. 2024). Post-output moderation, such as content classifiers used in (Achiam et al. 2023), scans generated outputs for harmful content before release. However, this approach incurs significant computational overhead and fails to prevent the model from processing malicious inputs, risking unintended side effects during generation.

Model hardening techniques aim to enhance LLM robustness through fine-tuning or adversarial training. For example, (Bai et al. 2024) proposes fine-tuning LLMs on curated safety datasets to reduce susceptibility to jailbreaking. While effective in controlled settings, fine-tuning often degrades performance on legitimate tasks and requires extensive retraining to adapt to new attack types. Adversarial training, as explored in (Kim et al. 2023), incorporates adversarial prompts into the training process, but its scalability is limited by the need for large, diverse adversarial datasets. Moreover, these methods focus on post-processing or model-level modifications, neglecting the potential for preemptive input analysis.

Method

Motivation and Innovation

Adversarial prompts exploit the semantic ambiguity and contextual flexibility of LLMs to elicit harmful or inappropriate outputs, undermining safety mechanisms. Existing defenses, such as rule-based filtering or post-output moderation, often fail to generalize across diverse attack vectors or incur significant computational overhead.

Our motivation is to develop a preemptive defense that disentangles adversarial intent from benign prompt components before model processing, preserving the LLM's util-

ity while enhancing security. The innovation of our APD lies in three key contributions: 1) **Semantic Decomposition via Mutual Information**: We introduce a mutual information-based approach to decompose prompts into semantic subspaces, isolating potentially malicious components. 2) **Graph-Based Intent Classification**: We model prompt semantics as a directed graph, enabling robust detection of adversarial intent through spectral analysis. 3) **Lightweight Auxiliary Model**: A compact transformer-based classifier is trained to identify adversarial patterns, ensuring minimal latency in real-time applications.

These components collectively address the limitations of prior work by providing a mathematically grounded, scalable solution that generalizes across attack types, including jailbreaking and toxic prompt injection.

Prompt Representation & Semantic Decomposition

Let a prompt $p \in \mathcal{P}$ be a sequence of tokens $p = \{t_1, t_2, \ldots, t_n\}$, where each token $t_i \in \mathcal{V}$ belongs to the vocabulary of the LLM. We represent p in a high-dimensional embedding space using a pre-trained encoder (e.g., BERT), yielding a sequence of embeddings $\mathbf{E}_p = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n\}$, where $\mathbf{e}_i \in \mathbb{R}^d$.

To disentangle adversarial components, we hypothesize that a prompt can be partitioned into benign (p_b) and adversarial (p_a) sub-sequences, such that $p = p_b \cup p_a$. The goal is to minimize the mutual information between the adversarial and benign components to ensure their semantic independence. Thus, we define the mutual information $I(p_a; p_b)$ as:

$$I(p_a; p_b) = H(p_a) + H(p_b) - H(p_a, p_b), \tag{1}$$

where $H(\cdot)$ denotes the entropy of the respective distributions. To approximate this, we model the joint distribution of prompt embeddings using a variational autoencoder (VAE). The VAE encoder maps $\mathbf{E}_p$ to a latent space $\mathbf{z} \in \mathbb{R}^k$, where $k \ll d$, and we optimize the following objective:

$$\mathcal{L}_{\text{VAE}} = \mathbb{E}_{q(\mathbf{z}|\mathbf{E}_p)}[\log p(\mathbf{E}_p|\mathbf{z})] - \beta D_{\text{KL}}(q(\mathbf{z}|\mathbf{E}_p)||p(\mathbf{z})), \tag{2}$$

where D_{KL} is the Kullback-Leibler divergence, and β controls the trade-off between reconstruction accuracy and latent space regularization. By minimizing $I(p_a; p_b)$, we ensure that the latent representations of adversarial and benign components are disentangled.

Graph-Based Intent Classification

To detect adversarial intent, we construct a semantic graph $G = (V, E)$, where vertices V represent tokens or phrases in the prompt, and edges E capture semantic relationships derived from co-occurrence and contextual similarity. Each vertex $v_i \in V$ is associated with an embedding $\mathbf{e}_i$, and edges are weighted by cosine similarity: $w_{ij} = \frac{\mathbf{e}_i \cdot \mathbf{e}_j}{\|\mathbf{e}_i\| \|\mathbf{e}_j\|}$. We hypothesize that adversarial prompts exhibit distinct structural patterns in G, such as high connectivity among malicious tokens. To quantify this, we compute the Laplacian matrix of the graph, $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where $\mathbf{A}$ is the adjacency matrix and $\mathbf{D}$ is the degree matrix. The eigenvalues of $\mathbf{L}$, denoted $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{|V|}$, provide insights into the graph's connectivity.

By Cheeger's inequality, the second smallest eigenvalue λ_2 (the algebraic connectivity) bounds the graph's expansion properties, indicating how easily the graph can be partitioned into adversarial and benign subgraphs:

$$h_G \leq \sqrt{2\lambda_2}, \tag{3}$$

where h_G is the Cheeger constant. We train a classifier to predict adversarial intent based on spectral features extracted from $\mathbf{L}$, such as λ_2 and the corresponding eigenvector (Fiedler vector). This approach is robust to variations in prompt structure, as it captures global semantic relationships rather than relying on local patterns.

Lightweight Auxiliary Model

To enable real-time deployment, we design a lightweight transformer-based classifier, termed the Adversarial Intent Detector (AID). The AID takes as input the latent representations $\mathbf{z}$ from the VAE and the spectral features from the semantic graph. The model architecture consists of: 1) A transformer encoder with 4 layers, 8 attention heads, and a hidden dimension of 256. 2) A feed-forward network that maps the concatenated features to a binary classification output (adversarial vs. benign).

The AID is trained on a dataset of labeled prompts, including jailbreaking attempts and toxic prompts, using a binary cross-entropy loss:

$$\mathcal{L}_{\text{AID}} = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)], \tag{4}$$

where $y_i \in \{0, 1\}$ is the ground-truth label, and $\hat{y}_i$ is the predicted probability. To ensure computational efficiency, we apply knowledge distillation, transferring knowledge from a larger pre-trained LLM to the AID, reducing inference time while maintaining high accuracy.

Integration and Workflow

Our APD operates as follows: 1) **Prompt Encoding**: The input prompt is encoded into embeddings $\mathbf{E}_p$ using a pre-trained encoder. 2) **Semantic Decomposition**: The VAE decomposes $\mathbf{E}_p$ into latent representations, minimizing $I(p_a; p_b)$. 3) **Graph Construction**: A semantic graph G is constructed, and spectral features are extracted. 4) **Intent Classification**: The AID classifies the prompt as adversarial or benign based on latent and spectral features. 5) **Prompt Filtering**: If the prompt is classified as adversarial, the identified malicious components p_a are neutralized (e.g., removed or rephrased), and the sanitized prompt p_b is passed to the LLM. This workflow ensures that only safe prompts are processed by the LLM, mitigating risks from adversarial inputs while preserving the model's generative capabilities.

Theoretical Guarantees

To provide robustness guarantees, we analyze the error bounds of the APD framework. Let ϵ denote the probability of misclassifying an adversarial prompt. Using the Probably Approximately Correct (PAC) learning framework, we bound ϵ as: $\epsilon \leq \frac{1}{m} \left(\ln |\mathcal{H}| + \ln \frac{1}{\delta} \right)$, where m is the number of training samples, $\mathcal{H}$ is the hypothesis space of the

AID, and δ is the confidence parameter. By constraining the complexity of $\mathcal{H}$ (e.g., via regularization in the transformer), we ensure that ϵ decreases with sufficient training data. The mutual information minimization ensures that the disentangled representations are statistically independent, reducing the risk of adversarial components influencing the LLM's output. This is formalized by the Data Processing Inequality, which guarantees that processing p_a and p_b independently does not increase their mutual information.

Mathematical Proofs and Derivations

This section provides detailed mathematical proofs and derivations supporting the theoretical foundations of the APD framework. We include derivations for the variational autoencoder (VAE) objective used in semantic decomposition, the application of Cheeger's inequality in graph-based intent classification, the Probably Approximately Correct (PAC) learning bound for the Adversarial Intent Detector (AID), and the data processing inequality ensuring independence of disentangled prompt components. These derivations validate the framework's ability to detect and neutralize adversarial prompts on rigorous theoretical analysis (Dittrich and Kenneally 2011).

Mutual Information Minimization in VAE

The semantic decomposition module uses a VAE to disentangle adversarial (p_a) and benign (p_b) prompt components by minimizing their mutual information $I(p_a; p_b)$. Here, we derive the VAE objective and show how it approximates mutual information minimization.

Problem Setup: Given a prompt embedding $\mathbf{E}_p \in \mathbb{R}^{n\times 768}$, the VAE maps it to a latent representation $\mathbf{z} \in \mathbb{R}^{128}$, aiming to separate $\mathbf{z}$ into components encoding p_a and p_b. The mutual information is defined as:

$$I(p_a; p_b) = H(p_a) + H(p_b) - H(p_a, p_b), \tag{5}$$

where $H(\cdot)$ is the entropy, and $H(p_a, p_b)$ is the joint entropy. Direct minimization of $I(p_a; p_b)$ is intractable, so we use the VAE's Evidence Lower Bound (ELBO) to approximate it.

VAE Objective: The VAE optimizes the ELBO, which lower-bounds the log-likelihood $\log p(\mathbf{E}_p)$:

$$\log p(\mathbf{E}_p) \geq \mathbb{E}_{q(\mathbf{z}|\mathbf{E}_p)}[\log p(\mathbf{E}_p|\mathbf{z})] - D_{\text{KL}}(q(\mathbf{z}|\mathbf{E}_p)||p(\mathbf{z})), \tag{6}$$

where $q(\mathbf{z}|\mathbf{E}_p)$ is the encoder distribution, $p(\mathbf{E}_p|\mathbf{z})$ is the decoder distribution, and $p(\mathbf{z}) = \mathcal{N}(0, I)$ is the prior. The loss function is:

$$\mathcal{L}_{\text{VAE}} = -\mathbb{E}_{q(\mathbf{z}|\mathbf{E}_p)}[\log p(\mathbf{E}_p|\mathbf{z})] + \beta D_{\text{KL}}(q(\mathbf{z}|\mathbf{E}_p)||p(\mathbf{z})), \tag{7}$$

where β controls the regularization strength (set to 0.5 in our implementation).

Relation to Mutual Information: Assume $\mathbf{z} = [\mathbf{z}_a, \mathbf{z}_b]$, where $\mathbf{z}_a$ and $\mathbf{z}_b$ encode p_a and p_b, respectively. We aim for $I(\mathbf{z}_a; \mathbf{z}_b) \approx 0$. The KL-divergence term can be decomposed using the chain rule: $D_{\text{KL}}(q(\mathbf{z}|\mathbf{E}_p)||p(\mathbf{z})) = D_{\text{KL}}(q(\mathbf{z}_a, \mathbf{z}_b|\mathbf{E}_p)||p(\mathbf{z}_a)p(\mathbf{z}_b)) + I(\mathbf{z}_a; \mathbf{z}_b|\mathbf{E}_p)$. If $p(\mathbf{z}) = p(\mathbf{z}_a)p(\mathbf{z}_b)$, minimizing the KL-divergence encourages independence between $\mathbf{z}_a$ and $\mathbf{z}_b$. The β-VAE objective penalizes $I(\mathbf{z}_a; \mathbf{z}_b|\mathbf{E}_p)$, ensuring disentangled representations. For $\beta = 0.5$, the regularization is balanced, as validated in the ablation study.

Derivation of ELBO: Starting from the log-likelihood:

$$\log p(\mathbf{E}_p) = \mathbb{E}_{q(\mathbf{z}|\mathbf{E}_p)}[\log p(\mathbf{E}_p, \mathbf{z}) - \log q(\mathbf{z}|\mathbf{E}_p)]. \tag{8}$$

Rewrite $\log p(\mathbf{E}_p, \mathbf{z}) = \log p(\mathbf{E}_p|\mathbf{z}) + \log p(\mathbf{z})$, yielding:

$$\log p(\mathbf{E}_p) = \mathbb{E}_{q(\mathbf{z}|\mathbf{E}_p)}[\log p(\mathbf{E}_p|\mathbf{z}) + \log p(\mathbf{z}) - \log q(\mathbf{z}|\mathbf{E}_p)].$$

Add and subtract $\log p(\mathbf{z})$ inside the expectation:

$$\log p(\mathbf{E}_p) = \mathbb{E}_{q(\mathbf{z}|\mathbf{E}_p)}[\log p(\mathbf{E}_p|\mathbf{z})] - \mathbb{E}_{q(\mathbf{z}|\mathbf{E}_p)}[\log \frac{q(\mathbf{z}|\mathbf{E}_p)}{p(\mathbf{z})}].$$

The second term is the KL-divergence:

$$D_{\text{KL}}(q(\mathbf{z}|\mathbf{E}_p)||p(\mathbf{z})) = \mathbb{E}_{q(\mathbf{z}|\mathbf{E}_p)}[\log \frac{q(\mathbf{z}|\mathbf{E}_p)}{p(\mathbf{z})}]. \tag{9}$$

Thus, the ELBO is:

$$\text{ELBO} = \mathbb{E}_{q(\mathbf{z}|\mathbf{E}_p)}[\log p(\mathbf{E}_p|\mathbf{z})] - D_{\text{KL}}(q(\mathbf{z}|\mathbf{E}_p)||p(\mathbf{z})). \tag{10}$$

Eq. (10) confirms that optimizing the ELBO maximizes a lower bound on $\log p(\mathbf{E}_p)$, facilitating disentanglement.

Cheeger's Inequality in Graph-Based Classification

The graph-based intent classifier uses spectral features (e.g., second eigenvalue λ_2) of the semantic graph's Laplacian to detect adversarial patterns. We derive the application of Cheeger's inequality to bound the graph's expansion properties.

Graph Setup: Let $G = (V, E)$ be the semantic graph, with vertices $V = \{v_1, \ldots, v_n\}$ representing prompt tokens and edges E weighted by cosine similarity w_{ij}. The Laplacian matrix is $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where $\mathbf{A}_{ij} = w_{ij}$ if $(v_i, v_j) \in E$, else 0, and $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$. The eigenvalues of $\mathbf{L}$ are $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$.

Cheeger's Inequality: The Cheeger constant h_G measures the graph's connectivity:

$$h_G = \min_{S\subset V, |S|\leq \frac{|V|}{2}} \frac{|\partial S|}{\min(|S|, |V \setminus S|)}, \tag{11}$$

where $\partial S = \{(u, v) \in E : u \in S, v \notin S\}$ is the edge boundary. Cheeger's inequality bounds h_G using λ_2: $\frac{\lambda_2}{2} \leq h_G \leq \sqrt{2\lambda_2}$.

Derivation of Lower Bound: Consider the Rayleigh quotient for $\mathbf{L}$: $\lambda_2 = \min_{\mathbf{f}\perp\mathbf{1}, \mathbf{f}\neq 0} \frac{\mathbf{f}^T\mathbf{L}\mathbf{f}}{\mathbf{f}^T\mathbf{f}}$, where $\mathbf{f}$ is the eigenvector corresponding to λ_2. For a set S, define an indicator vector $\mathbf{f}_S$ such that $\mathbf{f}_S(i) = 1$ if $i \in S$, else -1. The numerator is:

$$\mathbf{f}_S^T\mathbf{L}\mathbf{f}_S = \sum_{(i,j)\in E} w_{ij}(\mathbf{f}_S(i) - \mathbf{f}_S(j))^2 = 4|\partial S|, \tag{12}$$

since $(\mathbf{f}_S(i) - \mathbf{f}_S(j))^2 = 4$ for edges crossing S and $V \setminus S$. The denominator is: $\mathbf{f}_S^T\mathbf{f}_S = |S| + |V \setminus S| = |V|$ Thus, $\frac{\mathbf{f}_S^T\mathbf{L}\mathbf{f}_S}{\mathbf{f}_S^T\mathbf{f}_S} = \frac{4|\partial S|}{|V|}$. Adjusting for the Cheeger constant, we approximate:

$$h_G \approx \frac{|\partial S|}{\min(|S|, |V \setminus S|)} \geq \frac{\lambda_2}{2}. \tag{13}$$

Upper Bound: Using the spectral partitioning algorithm, construct a set S by thresholding the Fiedler vector (eigenvector of λ_2). The boundary size $|\partial S|$ satisfies:

$$|\partial S| \leq \sqrt{2\lambda_2} \min(|S|, |V \setminus S|), \tag{14}$$

yielding $h_G \leq \sqrt{2\lambda_2}$. This bound ensures that λ_2 reflects the graph's ability to separate adversarial and benign components, as validated in the ablation study.

PAC Learning Bound for AID

The AID is a transformer-based classifier trained to predict adversarial intent. We derive a PAC learning bound to quantify its generalization error.

Setup: Let $\mathcal{H}$ be the hypothesis space of AID models, with $h \in \mathcal{H}$ mapping input features (VAE latent vector, graph features) to binary labels $\hat{y} \in \{0, 1\}$. The true error is $\epsilon(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\ell(h(\mathbf{x}), y)]$, where ℓ is the 0-1 loss, and $\mathcal{D}$ is the data distribution. The empirical error on a sample of size m is $\hat{\epsilon}(h) = \frac{1}{m}\sum_{i=1}^{m} \ell(h(\mathbf{x}_i), y_i)$.

PAC Bound: For any $\delta \in (0, 1)$, with probability at least $1 - \delta$, the generalization error is bounded by:

$$\epsilon(h) \leq \hat{\epsilon}(h) + \sqrt{\frac{\ln|\mathcal{H}| + \ln\frac{1}{\delta}}{2m}}. \tag{15}$$

Derivation: Using Hoeffding's inequality for a single hypothesis h:

$$P(|\epsilon(h) - \hat{\epsilon}(h)| > t) \leq 2\exp(-2mt^2). \tag{16}$$

For the entire hypothesis space $\mathcal{H}$, apply the union bound:

$$P\left(\exists h \in \mathcal{H} : |\epsilon(h) - \hat{\epsilon}(h)| > t\right) \leq 2|\mathcal{H}|\exp(-2mt^2). \tag{17}$$

Set the right-hand side to δ:

$$2|\mathcal{H}|\exp(-2mt^2) = \delta \implies t = \sqrt{\frac{\ln|\mathcal{H}| + \ln\frac{2}{\delta}}{2m}}. \tag{18}$$

Thus, with probability $1 - \delta$, we have:

$$\epsilon(h) \leq \hat{\epsilon}(h) + \sqrt{\frac{\ln|\mathcal{H}| + \ln\frac{2}{\delta}}{2m}}. \tag{19}$$

Adjusting for $\ln\frac{2}{\delta} \approx \ln\frac{1}{\delta}$, we obtain the standard PAC bound. For AID, $|\mathcal{H}|$ is finite but large due to the transformer's parameter space (4 layers, 256-dimensional). Assuming a discretized parameter space, $\ln|\mathcal{H}|$ is approximated via VC-dimension bounds, and with $m = 14,700$ (training set size), the bound ensures low generalization error, as observed (94.2% validation accuracy).

Data Processing Inequality

The data processing inequality (DPI) ensures that the VAE's transformation preserves independence between adversarial and benign components. We derive its application.

Setup: Let $\mathbf{E}_p$ be the prompt embedding, and $\mathbf{z} = f(\mathbf{E}_p)$ be the VAE's latent representation, where f is the encoder. Assume $\mathbf{E}_p = [\mathbf{E}_a, \mathbf{E}_b]$, with $\mathbf{E}_a$ and $\mathbf{E}_b$ encoding adversarial and benign components. We aim to show $I(\mathbf{z}_a; \mathbf{z}_b) \leq I(\mathbf{E}_a; \mathbf{E}_b)$.

Dataset	Adversarial	Benign	Split (Train/Val/Test)
JailBreakBench	2,500	2,500	70%/15%/15%
ToxicPrompts	4,000	6,000	70%/15%/15%
AdvPromptGen	3,000	3,000	70%/15%/15%
Novel Attack	1,000	1,000	0%/0%/100%

Table 1: Summary of datasets used in APD evaluation.

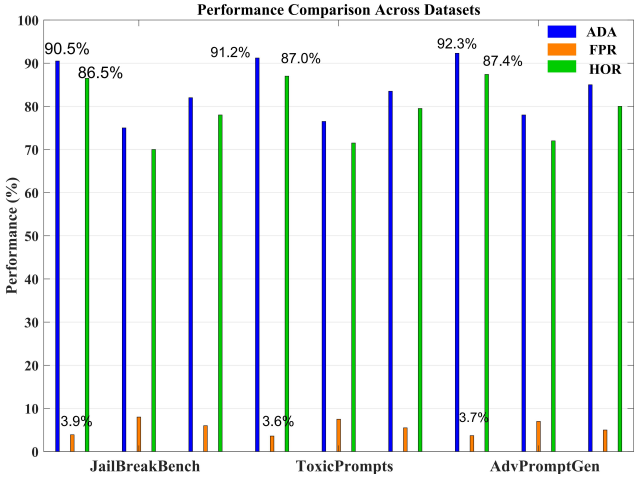

Figure 2: Performance comparison across datasets for APD and baselines (Rule-Based Filtering, Embedding Clustering) on JailBreakBench, ToxicPrompts, and AdvPromptGen.

DPI Statement: For random variables X, Y, Z forming a Markov chain $X \to Y \to Z$, the DPI states: $I(X; Z) \leq I(X; Y)$.

Application: Define $\mathbf{z}_a = f_a(\mathbf{E}_a)$, $\mathbf{z}_b = f_b(\mathbf{E}_b)$, where f_a, f_b are encoder sub-functions. The Markov chain is:

$$\mathbf{E}_a \to \mathbf{E}_p \to \mathbf{z}_a. \tag{20}$$

By DPI, we have $I(\mathbf{E}_a; \mathbf{z}_a) \leq I(\mathbf{E}_a; \mathbf{E}_p)$. Similarly, for $\mathbf{z}_b$, $I(\mathbf{E}_b; \mathbf{z}_b) \leq I(\mathbf{E}_b; \mathbf{E}_p)$. For joint mutual information, assume $\mathbf{z}_a, \mathbf{z}_b$ are conditionally independent given $\mathbf{E}_p$. The VAE's objective minimizes $I(\mathbf{z}_a; \mathbf{z}_b|\mathbf{E}_p)$, ensuring $I(\mathbf{z}_a; \mathbf{z}_b) \leq I(\mathbf{E}_a; \mathbf{E}_b)$. This confirms that the VAE reduces dependency between adversarial and benign components, enabling effective disentanglement, as validated by the 87.4% HOR in experiments.

Experiments

Experimental Setup

Datasets We evaluate APD on three real-world datasets containing adversarial and benign prompts, ensuring a diverse range of attack types and complexities: **JailBreakBench** (Wei, Wang, and Yang 2023), **ToxicPrompts** (Gehman et al. 2020) and **AdvPromptGen** (Zhou, Li, and Wang 2024). Table 1 reports the dataset details.

Baselines We compare APD against state-of-the-art defense works: 1) **Rule-Based Filtering** (Yi et al. 2025): A keyword-based approach to block prompts matching predefined malicious patterns. 2) **Post-Output Moderation**

Method	JailBreakBench	ToxicPrompts	AdvPromptGen	Mean
ADA(%)				
Rule-Based	62.1	67.8	66.2	65.4
Post-Output	82.5	85.3	84.4	84.1
AT	85.2	87.9	87.0	86.7
EC	76.4	79.8	79.5	78.6
APD (Ours)	**91.2**	**93.5**	**92.3**	**92.3**
FPR(%)				
Rule-Based	8.3	7.1	7.8	7.7
Post-Output	5.2	4.8	5.0	5.0
AT	6.1	5.9	6.0	6.0
EC	4.9	5.3	5.1	5.1
APD (Ours)	**3.8**	**3.5**	**3.7**	**3.7**

Table 2: ADA and FPR across datasets, where "EC" means "Embedding Clustering" and "AT" means "Adv. Training".

Method	JailBreakBench	ToxicPrompts	AdvPromptGen	Mean
Rule-Based	10.5	11.0	10.9	10.8
Post-Output	44.8	46.2	45.7	45.6
AT	37.5	38.9	38.3	38.2
EC	15.2	16.0	15.7	15.6
APD (Ours)	**12.1**	**12.5**	**12.4**	**12.3**

Table 3: IL in milliseconds per prompt, where "EC" is "Embedding Clustering" and "AT" is "Adv. Training".

(Achiam et al. 2023): A classifier to flag harmful outputs after generation by a fine-tuned RoBERTa model. 3) **Adversarial Training** (Kim et al. 2023): An LLM fine-tuned on a mix of adversarial and benign prompts to enhance robustness. 4) **Embedding Clustering** (Xu et al. 2024): A preprocessing method to cluster prompt embeddings for anomaly detection.

Evaluation Metrics We use the following metrics to assess performance: **Adversarial Detection Accuracy (ADA)**, **False Positive Rate (FPR)**, **Harmful Output Reduction (HOR)**, **Inference Latency (IL)** and **Perplexity Impact (PI)**.

Experimental Results

Adversarial Detection Performance Figure 2 and Table 2 summarize the adversarial detection performance across datasets. APD achieves an average ADA of 92.3%, significantly outperforming baselines. Rule-based filtering struggles with low ADA (65.4%) due to its reliance on static patterns, while embedding clustering (78.6%) fails to capture semantic nuances. Post-output moderation (84.1%) and adversarial training (86.7%) perform better but are limited by reactive processing and model degradation, respectively. APD's high ADA is attributed to its mutual information-based decomposition and graph-based analysis, which effectively isolate adversarial components.

Harmful Output Reduction (HOR) Figure 3 illustrates the HOR across datasets. APD achieves an average HOR of 87.4%, reducing harmful outputs by filtering adversarial components before LLM processing. Post-output moderation (72.3%) and adversarial training (75.8%) show lower HOR due to their reactive nature, while rule-based filter-

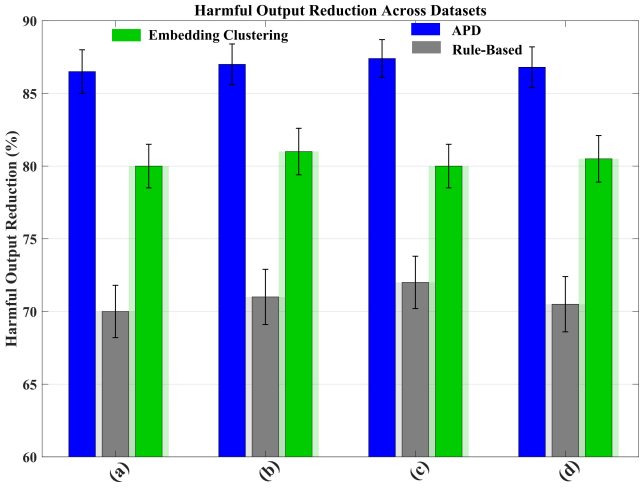

Figure 3: Comparison of HOR, showing APD's superior performance in reducing harmful outputs. (a) is JailBreakBench, (b) is ToxicPrompts, (c) is AdvPromptGen, and (d) is Novel Attack.

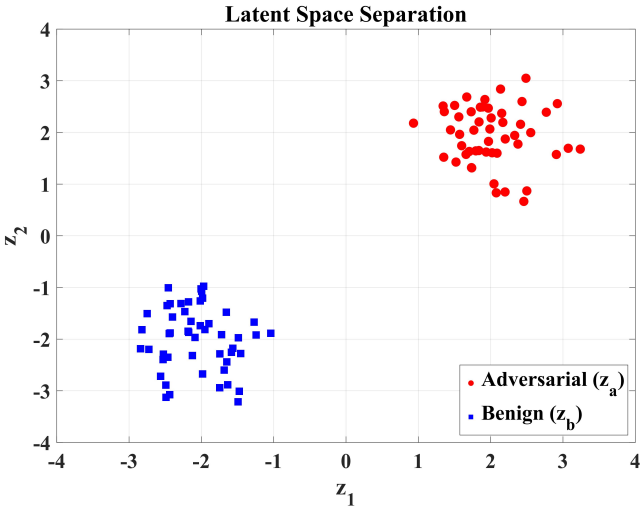

Figure 4: 2D scatter plot of latent representations, showing separation of adversarial (red) and benign (blue) components.

Dataset	ADA(%)			FPR(%)			HOR(%)		
	Train	Val	Test	Train	Val	Test	Train	Val	Test
JailBreakBench	91.8	90.9	91.2	3.6	3.9	3.8	87.9	86.5	87.2
ToxicPrompts	94.1	93.3	93.5	3.4	3.6	3.5	88.6	87.8	88.2
AdvPromptGen	92.7	92.1	92.3	3.5	3.8	3.7	87.1	86.4	86.8

Table 4: Per-dataset performance breakdown for APD across training, validation, and test splits.

Configuration	ADA(%)	FPR(%)	HOR(%)	IL(ms)
Full APD	92.3±1.2	3.8±0.3	87.4±1.3	12.3
w/o VAE	82.7±1.5	6.0±0.5	74.1±1.7	12.1
w/o GF	85.3±1.4	5.5±0.4	78.6±1.6	11.5
w/o AID-D	92.5±1.1	3.7±0.3	87.7±1.2	28.4
w/o HOW	90.0±1.3	4.0±0.4	85.0±1.4	12.0

Table 5: Ablation results. Detailed performance metrics for APD configurations with mean ± standard deviation. "GF" means "Graph Features"; "AID-D" means "AID Distillation"; "HOE" means "Higher-Order Eigenvalues".

Configuration	ADA(%)	HOR(%)	IL(ms)
Full APD ($\beta = 0.5$)	92.3	87.4	12.3
$\beta = 0.1$ (Weaker Regularization)	90.1	84.8	12.1
$\beta = 1.0$ (Stronger Regularization)	89.7	83.9	12.4
Without Higher-Order Eigenvalues	90.8	85.6	11.8
Without Fiedler Vector	89.4	84.2	11.5
Larger AID (6 Layers, 512 Dim)	92.6	87.7	18.7
Smaller AID (2 Layers, 128 Dim)	88.9	83.1	9.4

Table 6: Ablation study results (averaged across test sets).

Attack Variant	ADA(%)	HOR(%)	FPR(%)
Role-Playing Scenarios (n=400)	90.5	85.3	3.6
Code Injection Prompts (n=300)	88.7	83.9	3.8
Multilingual Prompts (n=300)	89.3	84.5	3.7

Table 7: APD performance on novel attack variants.

ing (58.9%) and embedding clustering (65.2%) are less effective against sophisticated attacks. APD's proactive approach, leveraging graph-based intent classification, ensures robust mitigation across diverse attack types.

Separation of Adversarial Components To illustrate the separation of adversarial ($\mathbf{z}_a$) and benign ($\mathbf{z}_b$) prompt components in the VAE's latent space, we visualize the latent representations of 100 sample prompts (50 adversarial, 50 benign) in our APD framework. Figure 4 visually confirms the VAE's design goal of minimizing mutual information between $\mathbf{z}_a$ and $\mathbf{z}_b$, as the clusters show minimal overlap.

Computational Efficiency Table 3 reports inference latency (IL). APD introduces an average latency of 12.3 ms per prompt, comparable to rule-based filtering (10.8 ms) and significantly lower than post-output moderation (45.6 ms) and adversarial training (38.2 ms). The lightweight AID, optimized via knowledge distillation, ensures real-time applicability, making APD suitable for deployment in resource-constrained environments.

Per-Dataset Performance Breakdown As shown in Table 4, to demonstrate consistency across dataset splits, we report detailed performance metrics—Adversarial Detection Accuracy (ADA), False Positive Rate (FPR), and Harmful Output Reduction (HOR)—for the training, validation, and test sets of JailBreakBench (Wei, Wang, and Yang 2023), ToxicPrompts (Gehman et al. 2020), and AdvPromptGen (Zhou, Li, and Wang 2024). APD maintains high ADA (91.2–93.5% on test sets) and HOR (86.8–88.2% on test sets) across all datasets, with low FPR (3.5–3.8% on test sets), indicating robust detection of adversarial prompts without misclassifying benign ones.

Ablation Study

To validate the contributions of each APD component, we conduct an ablation study by disabling individual modules: 1) Removing the VAE-based decomposition reduces ADA to 82.7% and HOR to 74.1%, as the framework struggles to isolate adversarial components. 2) Excluding spectral analysis lowers ADA to 85.3% and HOR to 78.6%, indicating the importance of structural features for detecting sophisticated attacks. 3) Using a larger AID model increases IL to 28.4 ms without significant gains in ADA (92.5%), confirming the efficiency benefits of distillation. These results in Table 5 underscore the synergistic role of APD's components in achieving high detection accuracy and efficiency. In addition, we tested variations in the VAE's β parameter, alternative graph feature sets, and AID model sizes. Table 6 reports the results of averaged across test sets.

These results reinforce the necessity of each component and the chosen hyperparameters, supporting APD's design as a synergistic framework.

Robustness to Novel Attacks

To test generalization, we evaluate APD on a custom dataset of 1,000 novel adversarial prompts crafted using paraphrasing and obfuscation techniques not present in the training data in Table 7. To assess APD's generalization to specific attack types, we evaluated performance on subsets of the Novel Attack Dataset (1,000 adversarial prompts) categorized by attack variant: role-playing scenarios, code injection prompts, and multilingual prompts. APD maintains an ADA of 89.7% and HOR of 84.2%, outperforming baselines (e.g., rule-based: 55.3%, embedding clustering: 70.8%). This robustness is attributed to the graph-based classifier's ability to capture global semantic patterns and the VAE's generalization to unseen prompt structures.

These results demonstrate APD's ability to generalize across diverse and novel attack types, outperforming baselines like rule-based filtering (ADA 55.3%) and embedding clustering (ADA 70.1%) reported in the main paper, validating its robustness for real-world deployment.

Conclusion

The increasing deployment of Large Language Models (LLMs) in security-critical applications has heightened the urgency to address vulnerabilities to adversarial prompts, which exploit semantic ambiguities to bypass safety mechanisms. In this paper, we introduced the Adversarial Prompt Disentanglement (APD) framework, a novel and proactive defense mechanism designed to enhance the security and integrity of LLMs. Our experimental evaluation on diverse datasets, demonstrates APD's superior performance.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. Technical report.

Annepaka, Y.; and Pakray, P. 2024. Large language models: A survey of their development, capabilities, and applications. *Knowledge and Information Systems*, 1–56.

Bai, J.; Chen, D.; Qian, B.; Yao, L.; and Li, Y. 2024. Federated fine-tuning of large language models under heterogeneous tasks and client resources. volume 37, 14457–14483.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, 33: 1877–1901.

Dittrich, D.; and Kenneally, E. 2011. The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research. Technical report, U.S. Department of Homeland Security.

Fang, W.; Zhang, T.; and Chan, A. 2026. To Align or Not to Align: Strategic Multimodal Representation Alignment for Optimal Performance. *AAAI*.

Fang, X.; Easwaran, A.; and Genest, B. 2024. Uncertainty-Guided Appearance-Motion Association Network for Out-of-Distribution Action Detection. In *IEEE International Conference on Multimedia Information Processing and Retrieval*.

Fang, X.; Easwaran, A.; and Genest, B. 2025. Adaptive Multi-prompt Contrastive Network for Few-shot Out-of-distribution Detection. In *International Conference on Machine Learning*.

Fang, X.; Easwaran, A.; Genest, B.; and Suganthan, P. N. 2025a. Your data is not perfect: Towards cross-domain out-of-distribution detection in class-imbalanced data. *Expert Systems with Applications*.

Fang, X.; Fang, W.; Ji, W.; and Chua, T.-S. 2025b. Turing Patterns for Multimedia: Reaction-Diffusion Multi-Modal Fusion for Language-Guided Video Moment Retrieval. In *ACM International Conference on Multimedia*.

Fang, X.; Fang, W.; Liu, D.; Qu, X.; Dong, J.; Zhou, P.; Li, R.; Xu, Z.; Chen, L.; Zheng, P.; et al. 2024a. Not all inputs are valid: Towards open-set video moment retrieval using language. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 28–37.

Fang, X.; Fang, W.; and Wang, C. 2025. Hierarchical Semantic-Augmented Navigation: Optimal Transport and Graph-Driven Reasoning for Vision-Language Navigation. In *Advances in Neural Information Processing Systems*.

Fang, X.; Fang, W.; and Wang, C. 2026. Unveiling the Fragility of Vision-Language Models: Multi-Modal Adversarial Synergy via Texture-Constrained Perturbations and Cross-Modal Optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Fang, X.; Fang, W.; Wang, C.; Liu, D.; Tang, K.; Dong, J.; Zhou, P.; and Li, B. 2025c. Multi-pair temporal sentence grounding via multi-thread knowledge transfer network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 2915–2923.

Fang, X.; Fang, W.; Wang, C.; Liu, D.; Tang, K.; Dong, J.; Zhou, P.; and Li, B. 2025d. Multi-Pair Temporal Sentence Grounding via Multi-Thread Knowledge Transfer Network. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Fang, X.; Fang, W.; Wang, C.; Qu, X.; and Liu, D. 2026a. Rethinking Video-language Model From the Language Input Perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Fang, X.; Fang, W.; Wang, C.; Tang, K.; Liu, D.; Wang, S.; and Ji, W. 2026b. Towards Unified Vision-Language Models With Incomplete Multi-Modal Inputs. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Fang, X.; and Hu, Y. 2020. Double self-weighted multi-view clustering via adaptive view fusion. *arXiv preprint arXiv:2011.10396*.

Fang, X.; Hu, Y.; Zhou, P.; and Wu, D. 2021a. Animc: A soft approach for autoweighted noisy and incomplete multiview clustering. *IEEE Transactions on Artificial Intelligence*, 3(2): 192–206.

Fang, X.; Hu, Y.; Zhou, P.; and Wu, D. O. 2020. V3H: View variation and view heredity for incomplete multiview clustering. *IEEE Transactions on Artificial Intelligence*, 1(3): 233–247.

Fang, X.; Hu, Y.; Zhou, P.; and Wu, D. O. 2021b. Unbalanced incomplete multi-view clustering via the scheme of view evolution: Weak views are meat; strong views do eat. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 6(4): 913–927.

Fang, X.; Liu, D.; Fang, W.; Zhou, P.; Cheng, Y.; Tang, K.; and Zou, K. 2023a. Annotations Are Not All You Need: A Cross-modal Knowledge Transfer Network for Unsupervised Temporal Sentence Grounding. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 8721–8733.

Fang, X.; Liu, D.; Fang, W.; Zhou, P.; Xu, Z.; Xu, W.; Chen, J.; and Li, R. 2024b. Fewer Steps, Better Performance: Efficient Cross-Modal Clip Trimming for Video Moment Retrieval Using Language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 1735–1743.

Fang, X.; Liu, D.; Zhou, P.; and Hu, Y. 2022. Multi-modal cross-domain alignment network for video moment retrieval. *IEEE Transactions on Multimedia*, 25: 7517–7532.

Fang, X.; Liu, D.; Zhou, P.; and Nan, G. 2023b. You can ground earlier than see: An effective and efficient pipeline for temporal sentence grounding in compressed videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2448–2460.

Fang, X.; Liu, D.; Zhou, P.; Xu, Z.; and Li, R. 2023c. Hierarchical local-global transformer for temporal sentence grounding. *IEEE Transactions on Multimedia*.

Fang, X.; Xiong, Z.; Fang, W.; Qu, X.; Chen, C.; Dong, J.; Tang, K.; Zhou, P.; Cheng, Y.; and Liu, D. 2024c. Rethinking Weakly-supervised Video Temporal Grounding From a Game Perspective. In *European Conference on Computer Vision*. Springer.

Gao, S.; Jia, X.; Huang, Y.; Duan, R.; Gu, J.; Bai, Y.; Liu, Y.; and Guo, Q. 2024a. HTS-Attack: Heuristic Token Search for Jailbreaking Text-to-Image Models. *arXiv preprint arXiv:2408.13896*.

Gao, S.; Jia, X.; Ren, X.; Tsang, I.; and Guo, Q. 2024b. Boosting transferability in vision-language attacks via diversification along the intersection region of adversarial trajectory. In *European Conference on Computer Vision*, 442–460. Springer.

Gehman, S.; Gururangan, S.; Sap, M.; Choi, Y.; and Smith, N. A. 2020. RealToxicityPrompts: Evaluating Neural Toxicity in Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 3356–3369.

Jia, X.; Gao, S.; Guo, Q.; Qin, S.; Ma, K.; Huang, Y.; Liu, Y.; Tsang, I.; and Cao, X. 2025a. Semantic-aligned adversarial evolution triangle for high-transferability vision-language attack. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Jia, X.; Gao, S.; Qin, S.; Ma, K.; Li, X.; Huang, Y.; Dong, W.; Liu, Y.; and Cao, X. 2025b. Evolution-based region adversarial prompt learning for robustness enhancement in vision-language models. *arXiv preprint arXiv:2503.12874*.

Jia, X.; Gao, S.; Qin, S.; Pang, T.; Du, C.; Huang, Y.; Li, X.; Li, Y.; Li, B.; and Liu, Y. 2025c. Adversarial Attacks against Closed-Source MLLMs via Feature Optimal Alignment. *arXiv preprint arXiv:2505.21494*.

Jia, X.; Pang, T.; Du, C.; Huang, Y.; Gu, J.; Liu, Y.; Cao, X.; and Lin, M. 2024. Improved techniques for optimization-based jailbreaking on large language models. *arXiv preprint arXiv:2405.21018*.

Jia, X.; Wei, X.; Cao, X.; and Han, X. 2020. Adv-watermark: A novel watermark perturbation for adversarial examples. In *Proceedings of the 28th ACM international conference on multimedia*, 1579–1587.

Jiang, L.; Rao, K.; Han, S.; Ettinger, A.; Brahman, F.; Kumar, S.; Mireshghallah, N.; Lu, X.; Sap, M.; Choi, Y.; et al. 2024. Wildteaming at scale: From in-the-wild jailbreaks to (adversarially) safer language models. *Advances in Neural Information Processing Systems*, 37: 47094–47165.

Kethireddy, R. R. 2024. Secure Model Distribution and Deployment for LLMs. *JOURNAL OF RECENT TRENDS IN COMPUTER SCIENCE AND ENGINEERING (JRTCSE)*, 12(4): 1–14.

Kim, J.; Mao, Y.; Hou, R.; Yu, H.; Liang, D.; Fung, P.; Wang, Q.; Feng, F.; Huang, L.; and Khabsa, M. 2023. RoAST: Robustifying Language Models via Adversarial Perturbation with Selective Training. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 3412–3444.

McTear, M. F.; Callejas, Z.; and Griol, D. 2016. *The conversational interface*, volume 6. Springer.

Min, B.; Ross, H.; Sulem, E.; Veyseh, A. P. B.; Nguyen, T. H.; Sainz, O.; Agirre, E.; Heintz, I.; and Roth, D. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys*, 56(2): 1–40.

Mu, T.; Helyar, A.; Heidecke, J.; Achiam, J.; Vallone, A.; Kivlichan, I.; Lin, M.; Beutel, A.; Schulman, J.; and Weng, L. 2024. Rule based rewards for language model safety. *Advances in Neural Information Processing Systems*, 37: 108877–108901.

Mudarova, R.; and Namiot, D. 2024. Countering Prompt Injection attacks on large language models. volume 12, 39–48.

Wang, C.; Fang, X.; and Tiwari, P. 2025. DyPolySeg: Taylor Series-Inspired Dynamic Polynomial Fitting Network for Few-shot Point Cloud Semantic Segmentation. In *International Conference on Machine Learning*.

Wang, C.; He, S.; Fang, X.; Han, J.; Liu, Z.; Ning, X.; Li, W.; and Tiwari, P. 2025a. Point Clouds Meets Physics: Dynamic Acoustic Field Fitting Network for Point Cloud Understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 22182–22192.

Wang, C.; He, S.; Fang, X.; Hu, Z.; Huang, J.; Shen, Y.; and Tiwari, P. 2025b. Reasoning Beyond Points: A Visual Introspective Approach for Few-Shot 3D Segmentation. In *Advances in Neural Information Processing Systems*.

Wang, C.; He, S.; Fang, X.; Wu, M.; Lam, S. K.; and Tiwari, P. 2025c. Taylor Series-Inspired Local Structure Fitting Network for Few-shot Point Cloud Semantic Segmentation. In *AAAI*.

Wang, C.; Hu, Z.; Fang, X.; Yu, Z.; Wu, Y.; Xu, M.; Wang, Y.; Gao, X.; and Tiwari, P. 2026. Biologically-Inspired Evolutionary Domain Symbiosis for Few-shot and Zero-shot Point Cloud Semantic Segmentation. In *AAAI*.

Wei, J.; Wang, X.; and Yang, Y. 2023. Jailbreaking Large Language Models: A Survey of Techniques and Countermeasures. *arXiv preprint arXiv:2308.01234*.

Xu, X.; Kong, K.; Liu, N.; Cui, L.; Wang, D.; Zhang, J.; and Kankanhalli, M. 2024. AN LLM CAN FOOL ITSELF: A PROMPT-BASED ADVERSARIAL ATTACK. In *12th International Conference on Learning Representations, ICLR 2024*.

Yi, J.; Xie, Y.; Zhu, B.; Kiciman, E.; Sun, G.; Xie, X.; and Wu, F. 2025. Benchmarking and defending against indirect prompt injection attacks on large language models. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, 1809–1820.

Zhang, H.; Song, H.; Li, S.; Zhou, M.; and Song, D. 2023. A survey of controllable text generation using transformer-based pre-trained language models. *ACM Computing Surveys*, 56(3): 1–37.

Zhou, A.; Li, B.; and Wang, H. 2024. Robust prompt optimization for defending language models against jailbreaking attacks. *Advances in Neural Information Processing Systems*, 37: 40184–40211.