# LLM SECURITY

## PROMPT INJECTION DETECTION


### Abstract

Large Language Models (LLMs) have transformed the field of natural language processing, showcasing remarkable capabilities in understanding and generating text across diverse applications. However, their pervasive use has exposed critical vulnerabilities, including prompt injection attacks, which can manipulate model responses and compromise their integrity. This research enhances existing solutions for detecting malicious prompts by building upon advancements in pre-trained Multilingual BERT embeddings [48]. We achieved a superior accuracy of 98.27% by leveraging Multilingual BERT for prompt tokenization and embedding generation, surpassing previous methodologies [48]. These embeddings were applied to train various machine learning models—such as Gaussian Naive Bayes, Random Forest, Support Vector Machine, and Logistic Regression—with Support Vector Machine demonstrating the most reliable performance. To strengthen the detection system further, we integrated heuristic-based anomaly detection for rapid identification of suspicious prompts and vector embedding similarity checks to capture semantically related adversarial inputs [15]. This combined framework addresses limitations of standalone methods, offering improved detection accuracy, robustness against evolving attack vectors, and scalability for real-world applications [49]. A detailed evaluation of misclassified cases provided insights for fine-tuning the model and enhancing its adaptability. By effectively mitigating the risks of prompt injection, this research contributes to the broader efforts in AI security, offering a practical and scalable defense mechanism to safeguard LLM applications.


## 1. Introduction

LLMs have become indispensable in modern applications, revolutionizing natural language processing with their ability to understand and generate human-like text. Despite their capabilities, their widespread adoption has also exposed critical vulnerabilities, particularly to adversarial attacks such as prompt injection and jailbreaking [16]. These threats can

manipulate LLM behavior, leading to unintended or harmful outcomes, thereby undermining their reliability and trustworthiness. In this chapter, we discuss the motivation for addressing these vulnerabilities, review existing works, outline the proposed solution, and highlight our contributions [47].

## 1.1 Motivation

The integration of LLMs into real-world systems has grown exponentially, encompassing domains such as customer support, education, healthcare, and content moderation. However, this rapid adoption brings security risks, with malicious actors exploiting vulnerabilities like prompt injection and alignment-breaking attacks to manipulate model outputs [8]. These attacks not only jeopardize data integrity and privacy but also raise ethical concerns regarding the misuse of AI systems [47].

Existing defenses often fail to address the evolving nature of these adversarial threats, especially in multi-modal and diverse real-world scenarios. This project is motivated by the pressing need to develop robust, scalable, and adaptable defense mechanisms to safeguard LLMs against malicious prompts and ensure their safe deployment [5].

## 1.2 Existing Work

Extensive research has been conducted to understand and mitigate the vulnerabilities in LLMs. Several notable studies have explored adversarial attacks and defenses:

- **Multi-Modal Vulnerabilities:** Research like *"Visual Adversarial Examples Jailbreak Large Language Models"* [1] and *"Jailbreak in Pieces: Compositional Adversarial Attacks on Multi-Modal Language Models"* [5] highlights the risks of adversarial inputs in multi-modal settings.
- **Prompt Injection Attacks:** Works such as *"Formalizing and Benchmarking Prompt Injection Attacks and Defenses"* [49] and *"Prompt Injection Attack Against LLM-integrated Applications"* [15] have laid the foundation for understanding text-based prompt injection and proposed initial defense strategies.
- **Alignment and Robustness:** Studies like *"Jailbroken: How Does LLM Safety Training Fail?"* [9] and *"Defending Against Alignment-Breaking Attacks via Robustly Aligned LLM"* [39] address the shortcomings of alignment techniques in preventing malicious outputs.
- **Embedding and Similarity Techniques:** The use of vector embeddings for semantic analysis, as explored in works like *"Backdooring Instruction-Tuned Large Language Models with Virtual Prompt Injection"* [36], has shown potential in detecting adversarial patterns.

While these studies provide valuable insights, gaps remain in creating unified frameworks that combine diverse detection and defense methods to address the complexity and dynamism of adversarial attacks.

### 1.3 Proposed Solution

Our research introduces a comprehensive, multi-layered framework to detect and mitigate malicious prompt injection attacks on LLMs. The key components of our proposed solution include:

- **Embedding-Based Detection:** Leveraging Multilingual BERT [48] to tokenize prompt texts and generate embeddings that enhance machine learning models' performance in binary classification tasks.
- **Heuristic-Based Detection:** Implementing a rule-based approach to flag suspicious prompts in real time based on predefined patterns and anomalies.
- **Similarity Checks Using Vector Embeddings:** Employing cosine similarity to compare input prompts with a reference dataset of known malicious and benign prompts, enabling semantic-level analysis and detection [49].
- **Unified Decision Framework:** Integrating these methods into a cohesive system that ensures robust, scalable, and real-time defense against prompt injection attacks.

This hybrid approach addresses the limitations of standalone methods, improving accuracy and adaptability in dynamic attack scenarios.

### 1.4 Our Contribution

This work extends the state-of-the-art in LLM security by achieving the following:

- **Enhanced Detection Accuracy:** Building upon *"Applying Pre-trained Multilingual BERT in Embeddings for Improved Malicious Prompt Injection Attacks Detection"* [48], we improved accuracy to 98.27%, outperforming previous works.
- **Comprehensive Framework:** Combining heuristic-based detection, embedding-based analysis, and similarity checks into a unified framework for robust adversarial defense.
- **Real-World Applicability:** Addressing both text-based and semantic-level threats, ensuring scalability and efficiency for deployment in real-world LLM-integrated systems.
- **Insightful Analysis:** Conducting detailed evaluations of misclassified prompts to identify weaknesses and refine detection models for future adversarial challenges.

This research not only strengthens the security of LLM applications but also provides a foundation for further exploration into multi-modal and adaptive adversarial defenses [5][36].

## 2. Background

LLMs have transformed the field of artificial intelligence with their remarkable ability to understand and generate human-like text. Despite their potential, their widespread adoption has raised significant security concerns.

### 2.1 LLM Architecture and Uses

LLMs are built on transformer architectures, which utilize self-attention mechanisms to process and generate language. The core of LLMs lies in their ability to learn patterns from vast amounts of text data during pre-training and adapt to specific tasks through fine-tuning or prompt engineering. Key architectural components include:

- **Self-Attention Mechanism:** Allows the model to focus on relevant parts of the input while processing sequential data.
- **Layer Stacking:** Stacks of transformer layers help in learning hierarchical representations of language.
- **Tokenization:** Breaks input text into smaller units, such as words or subwords, enabling efficient processing and embedding generation.
- **Embedding Layers:** Convert text into high-dimensional vector representations, capturing semantic meaning.

LLMs are widely used across various domains, including:

- Natural Language Understanding: Sentiment analysis, named entity recognition, and question answering.
- Content Generation: Writing assistance, code generation, and creative tasks like storytelling.
- Dialogue Systems: Chatbots, virtual assistants, and customer support systems.
- Decision Support Systems: Legal document analysis, financial forecasting, and medical diagnosis.

### 2.2 Security of LLM Models

As LLMs are integrated into sensitive applications, their security becomes paramount. These models face unique vulnerabilities due to their open-ended input nature and reliance on probabilistic outputs. Ensuring the security of LLMs involves identifying potential threats and implementing robust defense mechanisms.

#### 2.2.1 Threat Models

Threat models for LLMs describe scenarios where adversaries exploit system weaknesses to achieve malicious goals. Common threat models include:

- **Input Manipulation Attacks:** Adversaries craft inputs to alter model outputs, such as prompt injection or jailbreaking [8].
- **Output Exploitation:** Extracting sensitive information from models or generating harmful content.
- **Model Manipulation:** Using fine-tuning or backdoor attacks to insert malicious behaviors into the model.
- **Evasion Techniques:** Employing adversarial prompts that bypass safety filters to produce unsafe outputs [49].

**2.2.2 Input Manipulation Attacks:** There are majorly three types of input manipulation attacks however in this study we have mainly focused on Prompt Injection Attacks.

1. **Jailbreaking Attacks**
   Jailbreaking is a form of adversarial attack where carefully crafted prompts circumvent an LLM's alignment or safety filters, leading the model to generate restricted or harmful outputs [1][5][9][14][16][24]. Examples include:
   - *Compositional Jailbreaking:* Combining multiple benign prompts that, together, elicit restricted responses.
   - *Alignment Breaking:* Exploiting gaps in safety training to bypass ethical constraints.
   - *Stealthy Prompts:* Embedding malicious instructions in seemingly harmless inputs.

2. **Prompt Injection Attacks**
   Prompt injection manipulates LLMs by inserting adversarial instructions that override intended behaviors. These attacks exploit the model's tendency to prioritize the latest or most explicit instruction in the input.
   - *Direct Prompt Injection:* Explicitly inserting harmful commands into the input [36][37].
   - *Indirect Prompt Injection:* Embedding adversarial instructions within external data, such as documents or web content, that the model processes [47] [49].

3. **Multi-Modal Attacks**
   Multi-modal attacks target models capable of processing text along with other inputs like images, audio, or video [3][6]. These attacks exploit vulnerabilities in integrating multi-modal data:
   - *Visual Adversarial Examples:* Manipulated images or videos that mislead models into generating incorrect or harmful responses.
   - *Indirect Instruction Injection:* Combining adversarial inputs across multiple modalities to confuse or manipulate the model.

## 3. Literature Review

| S.No | Name | Summary |
|---|---|---|
| 1 | “Visual Adversarial Examples Jailbreak Large Language Models",2023-06,AAAI(Oral) 24, multi-modal [1] | This paper explores the security risks and challenges associated with integrating vision into Large Language Models (LLMs), exemplified by Visual Language Model (VLMs) like Flamingo and GPT-4. The authors highlight two main concerns:<br>1. Expansion of Attack Surfaces: The addition of visual input increases the vulnerability of LLMs to adversarial attacks, as the visual domain is continuous and high- dimensional, making it easier for attackers to exploit. This contrasts with purely textual adversarial attacks, which are more challenging due to the discrete nature of text.<br>2. Extended Adversarial Objectives: LLMs’ versatility allows adversarial attacks to go beyond simple misclassification, enabling attackers to achieve a broader range of malicious goals, such as generating toxic content or bypassing safety mechanisms. |
| 2 | “Are aligned neural network adversarially aligned?",2023-0 NeurIPS(Poster) 23, multi-modal [2] | This paper investigates the vulnerabilities of aligned Large Language Models (LLMs) to adversarial inputs, specifically focusing on their susceptibility to "adversarial alignment." While these models are designed to be "helpful and harmless" through alignment techniques like reinforcement learning with human feedback (RLHF), adversarial users can craft inputs that bypass these defenses. The study finds that traditional NLP-based adversarial attacks are not powerful enough to consistently break the alignment of text-only models. However, brute force methods reveal that adversarial examples capabl of eliciting harmful behavior do exist, suggesting that |

| | | |
|---|---|---|
| | | current attacks are insufficient to assess the true robustness of these models. The findings call for further research into adversarial alignment, particularly in the context of multimodal models, to address the security risks posed by these vulnerabilities. The paper concludes that current alignment methods are insufficient to eliminate adversarial threats, urging the community to develop stronger defenses. |
| 3 | "(Ab)using Images and Sounds for Indirect Instruction Injection in Multi-Modal LLMs", 2023-07, mult modal [3] | This paper explores how adversarial perturbations in images and audio can be used for indirect prompt and instruction injection in multi-modal Large Language Models (LLMs). An attacker subtly modifies images or audio files by embedding adversarial prompts, which are not noticeable to the user. When the user inputs the modified content into an unaltered multi-modal LLM, the model is manipulated to follow the attacker's instructions or produce specific attacker-chosen text. This attack method is demonstrated with proof-of- concept examples targeting LLaVA and PandaGPT, two open-source multi-modal LLMs.<br>The paper identifies two main types of injection attacks:<br>1. Targeted-output attack: The LLM is forced to generate a specific output (e.g., a string chosen by the attacker) when asked about the adversarially perturbe input.<br><br>Dialog poisoning: This auto-regressive attack manipulates the LLM to inject instructions into the ongoing conversation, steering it towards attacker- defined goals by exploiting the model's use of conversation history. |

| | | |
|---|---|---|
| 4 | "Universal and Transferable Adversarial Attacks on Aligned Language Models", 2023-07, transfer [4] | This paper presents a new adversarial attack method that enables aligned Large Language Models (LLMs) to generate objectionable content by attaching an adversarial suffix to user queries. Unlike traditional jailbreaks that rely on manual crafting, this approach uses automated techniques—greedy and gradient- based search methods—to generate highly effective and transferable adversarial suffixes. These suffixes ar designed to maximize the probability of eliciting harmful or inappropriate behavior from a model, often starting with an affirmative response to a potentially harmful prompt. |
| 5 | “Jailbreak in pieces: Compositional Adversarial Attacks on Multi-Moda Language Models", 2023-07, multi modal | This paper introduces new cross-modality adversarial attacks targeting Vision Language Models (VLMs), whic are resistant to traditional text-based jailbreak attacks The authors develop a novel compositional strategy, pairing benign-looking adversarial images with generic textual prompts to bypass the alignment of the language model. By exploiting vulnerabilities in the vision-to-text alignment, the adversarial images guide the model’s response to harmful behaviors. The attack operates without access to the LLM, relying only on th vision encoder, such as CLIP, lowering the barrier for attackers, especially in closed-source models.<br>The attacks leverage embedding-space-based method utilizing gradients to update images so that they align with toxic embeddings. Four different triggers are used—textual, OCR textual, visual, and combined OCR visual—to conceal malicious prompts within images.<br>The compositional nature of the attack |

| | | |
|---|---|---|
| | | allows a single malicious image to activate various benign text instructions, or a single text instruction to pair with different malicious triggers. This approach differs from traditional fully-gradient-based methods by allowing more generalization and flexibility. |
| 6 | Image Hijacking: Adversarial Image can Control Generative Models at Runtime", 2023-09, multi-modal [6] | This paper investigates the security vulnerabilities of Vision-Language Models (VLMs) against adversarial attacks, focusing on the image input to such models. The authors introduce "image hijacks," adversarial images that can control VLM behavior at inference time. The key contributions include:<br>1,Behaviour Matching Algorithm: A method to train adversarial image hijacks that exhibit transferability to unseen user inputs. This leads to the development of Prompt Matching, allowing adversarial images to mimi arbitrary text prompts (e.g., making a VLM believe tha the Eiffel Tower is in Rome), using a generic dataset unrelated to the specific prompt.<br>Types of Attacks: The authors craft four image hijack scenarios: (i) forcing VLMs to generate arbitrary string<br>(ii) bypassing safety mechanisms (jailbreaking), (iii) causing VLMs to leak their input context, and (iv) making VLMs believe false information (disinformation).<br>Evaluation of Hijacks: The paper systematically evaluates these image hijacks using constraints like $\ell\infty$ norm and patch constraints. Results show that image hijacks outperform state-of-the-art text-based adversarial methods, achieving over 80% success acros various models like LLaVA (a CLIP |

| | | |
|---|---|---|
| | | and LLaMA-2-based VLM). |
| 7 | “Weak-to-Strong Jailbreaking on Large Language Models", 2024-04, token-prob [7] | Vulnerability to Jailbreaking: Aligned LLMs can still be compromised through adversarial prompts, tuning, or decoding methods, as indicated by red-teaming report Observation on Decoding Distributions: The authors note that the decoding distributions of jailbroken and aligned models differ primarily in their initial generations. This insight leads to the development of a new attack strategy.<br>Weak-to-Strong Jailbreaking Attack: This proposed attack allows adversaries to leverage smaller, less secure aligned LLMs (e.g., a 7 billion parameter model) to aid in jailbreaking larger, more secure aligned mode (e.g., a 70 billion parameter model). By decoding the smaller LLMs just twice, attackers can effectively guide the jailbreaking process, significantly reducing computational demands and latency compared to directly decoding the larger models. |
| 8 | “Not what you've signed up for: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection", 2023-0 AISec@CCS 23 [8] | The paper addresses the security vulnerabilities of Large Language Models (LLMs), particularly focusing o a novel attack vector called Indirect Prompt Injection (IPI). Here’s a concise summary of the key points:<br>Vulnerability to Attacks:<br>LLMs, like ChatGPT and GPT-4, can be manipulated through adversarial prompting, specifically via Prompt Injection (PI) attacks, which can override intended instructions and controls.<br>Introduction of Indirect Prompt Injection:<br>IPI allows adversaries to exploit LLM-integrated applications remotely by injecting malicious prompts into data |

| | | |
|---|---|---|
| | | that may be retrieved during inference, blurring the line between data and instructions.<br>Taxonomy of Threats:<br>The authors develop a comprehensive taxonomy to systematically analyze the impacts and vulnerabilities associated with IPI, including risks such as data theft, information contamination, and denial of service.<br>Demonstration of Practical Attacks:<br>The paper showcases the viability of these attacks against real-world systems (e.g., Bing's GPT-4) and synthetic applications, revealing how retrieved promp can manipulate model behavior and API interactions. Urgent Need for Mitigations:<br>The authors highlight the current lack of effective defenses against these emerging threats, advocating f increased awareness and the development of robust protective measures. |
| 9 | “Jailbroken: How Does LLM Safety Training Fail?", 2023-07, NeurIPS(Oral) 23 [9] | The paper "Jailbroken: How Does LLM Safety Training Fail?" (NeurIPS 2023) investigates why large language models (LLMs), despite safety training, are still vulnerable to jailbreak attacks that cause them to exhibit undesired or harmful behaviors. This research, conducted by Alexander Wei, Nika Haghtalab, and Jacob Steinhardt, explores two primary failure modes i the safety training of these models: competing objectives and mismatched generalization. |

| | | |
|---|---|---|
| 10 | "Latent Jailbreak: A Benchmark for Evaluating Text Safety and Output Robustness of Large Language Models", 2023-07 [10] | The paper "Latent Jailbreak: A Benchmark for Evaluating Text Safety and Output Robustness of Large Language Models" (2023) introduces a benchmark specifically designed to test the safety and robustness of large language models (LLMs) in response to jailbreak-style prompts. The researchers highlight that despite advancements in training techniques such as instruction tuning and reinforcement learning from human or AI feedback, LLMs remain vulnerable to certain types of "latent jailbreaks." These are indirect embedded malicious prompts that can bypass safety filters and result in harmful or unintended outputs. |
| 11 | “Effective Prompt Extraction from Language Models", 2023-07, prompt-extraction [11] | The paper "Effective Prompt Extraction from Language Models" (2023) delves into how attackers can systematically exploit language models to retrieve underlying prompts. The research highlights several prompt extraction attack strategies, which focus on th vulnerabilities of models like GPT-3.5, GPT-4, and Vicuna-13B. These models were tested with various datasets (e.g., ShareGPT), revealing that a significant percentage of prompts can indeed be extracted successfully, especially from GPT-3.5, where over 80% of prompts were retrieved. |

| | | |
|---|---|---|
| 12 | “Multi-step Jailbreaking Privacy Attacks on ChatGPT", 2023-04, EMNLP 23, privacy [12] | The paper "Multi-step Jailbreaking Privacy Attacks on ChatGPT" from EMNLP 2023 examines the privacy vulnerabilities in ChatGPT, particularly focusing on the risk of extracting personal information through multi- step jailbreaking techniques. The authors developed a series of multi-step attacks that combine jailbreak prompts with advanced extraction methods to target specific types of private information, like email conten or personally identifiable information (PII). These attacks leverage prompt engineering tactics that bypas standard safety protocols, achieving higher success rates in eliciting sensitive data from ChatGPT, especiall in older model versions such as GPT-3.5. |
| 13 | “LLM Censorship: A Machine Learning Challenge or a Computer Security Problem?", 2023-07 [13] | The paper titled "LLM Censorship: A Machine Learning Challenge or a Computer Security Problem?" discusses the challenges and limitations of implementing effective censorship mechanisms in large language models (LLMs). The authors explore the inadequacy of traditional machine learning-based censorship methods, which often focus on semantic filters to bloc undesired content. However, they argue that these approaches fall short because the problem may not be purely a machine learning challenge but a complex security issue. |

| | | |
|---|---|---|
| 14 | “Jailbreaking chatgpt via prompt engineering: An empirical study", 2023-05 [14] | The study titled "Jailbreaking ChatGPT via Prompt Engineering: An Empirical Study" explores the vulnerabilities of ChatGPT, specifically its susceptibilit to prompt engineering techniques that can bypass built- in content restrictions. Conducted by researchers from Nanyang Technological University and Virginia Tech, the paper systematically categorizes jailbreak prompts into ten distinct patterns and three broad types: Pretending, Attention Shifting, and Privilege Escalation |
| 15 | “Prompt Injection attack against LLM-integrated Applications", 2023-06 [15] | The paper "Prompt Injection Attack Against LLM- integrated Applications" investigates the security vulnerabilities associated with prompt injection attacks on applications that utilize Large Language Models (LLMs). The authors—Yi Liu and colleagues— highlight how the growing integration of LLMs into commercial applications can pose significant risks, as these models can be manipulated through cleverly crafted inputs.<br>The study begins with an exploratory analysis of ten commercial applications, identifying the limitations of existing attack methods. In response, the authors introduce a new attack technique named HouYi, inspire by traditional web injection attacks. HouYi consists of three components: a pre-constructed prompt, an injection prompt that creates a context partition, and a malicious payload to achieve the attacker's objectives. |

| | | |
|---|---|---|
| 16 | “MasterKey: Automated Jailbreak Across Multiple Large Language | The paper titled "MASTERKEY: Automated Jailbreak Across Multiple Large Language Model Chatbots" focuses on the vulnerabilities present in large language model (LLM) chatbot. |
| 17 | "GPT-4 Is Too Smart To Be Safe: Stealthy Chat with LLMs via Cipher", 2023-08, ICLR 24, cipher [17] | The paper titled "GPT-4 Is Too Smart To Be Safe: Stealthy Chat with LLMs via Cipher," presented at the International Conference on Learning Representations (ICLR) 2024, investigates vulnerabilities in the safety mechanisms of Large Language Models (LLMs), particularly GPT-4. The authors explore how communication through ciphers can bypass these safety features that are primarily designed for natural languag processing.<br>The researchers introduced a framework called CipherChat, which allows users to interact with LLMs using ciphered prompts. This approach tests the robustness of safety alignment by assessing LLMs across various safety domains in both English and Chinese. Their findings reveal that certain ciphers can consistently bypass the safety measures of GPT-4, raising concerns about the model's reliability in adherin to safety protocols when faced with non-standard input |
| 18 | “Use of LLMs for Illicit Purposes: Threats, Prevention Measures, and Vulnerabilities", 2023-08 [18] | The paper titled "Use of LLMs for Illicit Purposes: Threats, Prevention Measures, and Vulnerabilities," authored by Maximilian Mozes et al., provides a comprehensive overview of the potential misuse of large language models (LLMs) in illegal activities. The authors highlight the rapid development of LLMs and the associated security risks, including |

| | | |
|---|---|---|
| | | their potential use for fraud, impersonation, and generating malware.<br>The paper categorizes the threats posed by LLMs into a taxonomy that outlines their generative capabilities and discusses prevention measures aimed at mitigating these risks. It emphasizes the importance of raising awareness among developers and users about the limitations and vulnerabilities of LLMs, especially as they become more integrated into various applications. |
| 19 | “Do-Not-Answer: A Dataset for Evaluating Safeguards in LLMs", 2023-08 [19] | The paper "Do-Not-Answer: A Dataset for Evaluating Safeguards in LLMs" presents an open-source dataset aimed at evaluating the safety mechanisms of large language models (LLMs). The researchers recognize that as LLMs evolve, they may develop harmful capabilities that are difficult to predict, necessitating robust evaluation methods to identify potential risks before deployment.<br>Key features of the study include:<br>1. Dataset Composition: The dataset consists of 939 carefully curated instructions that LLMs should not follow, categorized into five risk areas and twelve harm types. This structure helps in assessing LLMs responses to potentially harmful queries.<br>2. Evaluation Methodology: The paper assesses the responses of six popular LLMs, including GPT-4 and ChatGPT, through both human and automatic evaluations. The evaluation involves determining whether the models' responses are harmful (binary classification) and categorizing the type of actions they take in |

| | | |
|---|---|---|
| | | response<br>3. Performance Metrics: The results show that a simpl BERT-style classifier can achieve safety evaluation results comparable to those from GPT-4, demonstrating the effectiveness of their dataset for automatic assessments<br>.Findings: The assessment indicates that most LLMs provide safe responses across the risk areas examined. LLaMA-2 performed the best in terms of harmlessness, followed closely by ChatGPT and Claude(X-MOL). |
| 20 | “Detecting Language Model Attac with Perplexity", 2023-08 [20] | The paper "Detecting Language Model Attacks with Perplexity," authored by Gabriel Alon and Michael J Kamfonas, addresses the emerging threat of adversarial suffix attacks on large language models (LLMs). These attacks involve appending specific strings of text to prompts to manipulate LLMs into generating harmful content, such as instructions for illegal activities.<br>Key findings from the study include:<br>1. High Perplexity as an Indicator: The researchers utilized perplexity, a common metric in naturallanguage processing that measures how predictable a piece of text is, to identify adversarial prompts.<br>They found that adversarial suffixes significantly increased the perplexity of prompts, often exceeding a threshold of 1000.<br>2. Challenges with False Positives: While perplexity proved useful for detection, the researchers noted that relying solely on this metric led to a high rate of false positives. To mitigate this, they combined |

| | | |
|---|---|---|
| | | perplexity with token sequence length, using a Light Gradient-Boosting Machine (LightGBM) for classification. This approach improved the accuracy of detecting adversarial prompts while reducing false alarms.<br>Dataset Construction: The study involved two datasets—one with machine-generated adversarial prompts and another with human-crafted prompts. The results highlighted the diverse characteristics of adversarial prompts, emphasizing the need for robust detection methods that can differentiate between benign and malicious intents |
| 21 | “Open Sesame! Universal Black Bo Jailbreaking of Large Langua Models", 2023-09, gene-algorithm<br>[21] | paper "Open Sesame! Universal Black Bo Jailbreaking of Large Language Models," presented ICLR 2024, explores a novel method for manipulatin large language models (LLMs) to elicit harmful undesirable outputs. The authors propose using a genet algorithm (GA) to create a universal adversarial prom that can disrupt the model's alignment with user intent an social guidelines, even under black box conditions whe the model's internal parameters are not accessible. |
| 22 | “Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To!", 2023-10, ICLR(oral) 24 [22] | The paper titled "Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To!" discusses the unintended consequences of customizing large language models (LLMs) through fine-tuning. Conducted by researchers from institutions like Princeton and Stanford, the study reveals that fine-tuning even with benign or commonly used datasets can significantly |

| | | |
|---|---|---|
| | | compromise the safety mechanisms embedded in these models.<br>Key findings include:<br>1. Safety Risks in Fine-tuning: The researchers found that it only takes a few adversarially designed training examples—sometimes as few as 10—to jailbreak models like OpenAI's GPT-3.5 Turbo, making them vulnerable to harmful requests. This process was inexpensive, costing less than $0.20<br>2. Benign Data Also Risks Safety: Even fine-tuning on datasets that are not explicitly harmful can degrade safety. For example, training on commonly used datasets inadvertently removed safety guardrails, leading to the models becoming more responsive to harmful instructions<br>3. Implications for Developers and Policymakers: The findings highlight the need for heightened awareness among developers and policymakers regarding the trade-off between model<br>customization and safety. It stresses the necessity for improved safety mechanisms during the fine- tuning process<br>4. Mitigation Strategies: The authors suggest several potential strategies to retain safety, such as filtering harmful training data, employing "self-destructing models," and enhancing detection of harmful outputs. However, they caution that no current strategy is foolproof. |

| | | |
|---|---|---|
| 23 | “AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Larg Language Models", 2023-10, ICLR(poster) 24, gene-algorithm, new-criterion [23] | The paper titled "AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models" introduces a novel approach for generating prompts designed to bypass safety measures in aligned large language models (LLMs). The authors highlight that existing jailbreak techniques either require extensive manual crafting or produce prompts that lack semantic meaning, making them easier to detect.<br>Key Contributions:<br>1. AutoDAN Framework: This framework uses a hierarchical genetic algorithm to automatically generate prompts that maintain semantic meaningfulness while successfully bypassing LLM restrictions.<br>Initialization and Optimization: The approach starts with handcrafted prompts that have proven effective and evolves them using a genetic algorithm. This dual-layer optimization helps in exploring a wider solution space while ensuring that the generated prompts are not too farremoved from the original effective prompts. 3. Performance Evaluation: The paper showcases AutoDAN's superior performance in cross-model transferability and universality compared to existing methods. It also demonstrates the ability to evade perplexity-based defenses effectively. |
| 24 | "Jailbreak and Guard Aligned Language Models with Only Few In-Context Demonstrations", 2023-10, CoRR 23, ICL[24] | The paper "Jailbreak and Guard Aligned Language Models with Only Few In-Context Demonstrations," authored by Zeming Wei, Yifei Wang, and Yisen Wang, explores the |

| | | |
|---|---|---|
| | | vulnerabilities of large language models (LLMs) to jailbreaking attacks and proposes methods to both exploit and defend against these attacks using In-Context Learning (ICL).<br>Key Findings:<br>1. In-Context Learning (ICL): The researchers discovered that LLMs can be manipulated to increase or decrease their susceptibility to jailbreaking through the use of few in-context demonstrations. By presenting specific examples within the prompt, the model's behavior can be guided to either produce harmful outputs (jailbreaking) or reject malicious prompts (guarding).<br>2. In-Context Attack (ICA): The paper introduces the concept of ICA, where adversarial demonstrations are used to encourage the model to generate harmful content. This method is<br>efficient, requiring only a few demonstrations to effectively modify the model's responses to harmful prompts.<br>3. In-Context Defense (ICD): Conversely, the ICD method aims to strengthen model defenses by providing examples that demonstrate refusal to engage with harmful content. This technique enhances the model's robustness against potential attacks.<br>4. Experimental Results: The authors conducted experiments using an open-source aligned model (Vicuna-7B) to evaluate the effectiveness of ICA and ICD. The results indicated a notable increase in the success rate of adversarial |

| | | |
|---|---|---|
| | | attacks with just one demonstration, showing a rising trend up to 44% with five demonstrations.<br>Implications: The findings highlight the dual- edged nature of ICL in LLMs. While it can be exploited to induce harmful outputs, it also offers a framework for improving model safety and alignment by carefully curating the in-context demonstrations used in prompts. |
| 25 | "Multilingual Jailbreak Challenges in Large Language Models", 2023- 10, ICLR(poster) 24[25] | The paper "Multilingual Jailbreak Challenges in Large Language Models," presented at ICLR 2024, addresses the safety issues of large language models (LLMs) in a multilingual context, focusing on how these models can be manipulated or "jailbroken" through non-English prompts. The authors explore two scenarios: unintentional and intentional.<br>Unintentional Scenario: This involves users querying LLMs in low-resource languages (languages with fewe training data). The study finds that as language resourc decrease, the likelihood of encountering unsafe content increases significantly. For instance, low-resource languages exhibit an unsafe content rate that is approximately three times higher than that of high- resource languages, with rates of unsafe outputs reaching as high as 55%.<br>Intentional Scenario: In this scenario, malicious users exploit the vulnerabilities of LLMs by combining harmful instructions with multilingual prompts. The study reveals alarmingly high rates of unsafe output under this scenario—up to 80.92% for ChatGPT and 40.71% for GPT-4 when using |

| | | |
|---|---|---|
| | | malicious multilingual queries. |
| 26 | "Scalable and Transferable Black-Box Jailbreaks for Language Mode via Persona Modulation", 2023-11, SoLaR(poster) 24,[26] | The paper titled "Scalable and Transferable Black- Box Jailbreaks for Language Models via Persona Modulation" explores vulnerabilities in large language models (LLMs) like GPT-4, Claude 2, and Vicuna by using a technique called persona modulation. This method allows attackers to manipulate the models into adopting personas that are more likely to comply with harmful instructions.<br>Key highlights from the research include:<br>1. Automation of Attacks: The authors developed a framework that automates the generation of jailbreaking prompts using a language model assistant, making it easier to create effective attacks. This significantly reduces the time and effort required compared to manual prompt crafting.<br>2. Harmful Output Rates: The study found that employing persona modulation led to a harmful completion rate of 42.5% for GPT-4, a drastic increase from 0.23% without modulation. The harmful completion rates for Claude 2 and Vicuna were 61.0% and 35.9%, respectively.<br>3. Types of Harmful Instructions: The paper documents various harmful outputs generated through these attacks, including instructions for illegal activities like synthesizing drugs and money laundering.<br>Transferability of Attacks: The persona- modulation prompts were not only effective against GPT-4 but also successfully transferred to other |

| | | |
|---|---|---|
| | | models, indicating a broader vulnerability across LLMs. |
| 27 | "DeepInception: Hypnotize Large Language Model to Be Jailbreaker" 2023-11[27] | The paper "DeepInception: Hypnotize Large Language Model to Be Jailbreaker" proposes a novel approach to jailbreak large language models (LLMs) by exploiting their personification capabilities. The authors draw inspiration from the Milgram experiment, which demonstrated how individuals can be influenced to act against their ethical beliefs under authoritative instructions. This method, called DeepInception, involves constructing a layered scenario that subtly guides the model into generating harmful content without directly confronting its safety constraints. |
| 28 | "A Wolf in Sheep's Clothing: Generalized Nested Jailbreak Prompts can Fool Large Language Models Easily", 2023-11, NAACL 24[28] | The paper titled "A Wolf in Sheep's Clothing: Generalized Nested Jailbreak Prompts can Fool Large Language Models Easily" explores vulnerabilities in Large Language Models (LLMs) like ChatGPT and GPT-4, specifically focusing on how adversarial prompts known as "jailbreaks" can bypass safety measures. The authors, Peng Ding et al., present an automatic framework called ReNeLLM that generates effective jailbreak prompts by utilizing techniques like prompt rewriting and scenario nesting.<br>The study highlights the limitations of existing jailbreak methods, which often require intricate manual design or optimization processes that hinder generalization and efficiency. ReNeLLM aims to automate these processes, improving the success rate |

| | | |
|---|---|---|
| | | of attacks while reducing time costs compared to previous methods. The paper also critiques the current defense mechanisms in LLMs and proposes new strategies to enhance their safety against such attacks |
| 29 | "AutoDAN: Automatic and Interpretable Adversarial Attacks o Large Language Models", 2023-10 [29] | The paper titled "AutoDAN: Automatic and Interpretable Adversarial Attacks on Large Language Models" presents a novel approach to adversarial attacks targeting large language models (LLMs). The authors, Sicheng Zhu et al., highlight the ongoing vulnerability of LLMs to various jailbreak attacks, which can compromise their safety protocols.<br>Key Contributions:<br>1. Interpretability: AutoDAN generates interpretable attack prompts that resemble manual jailbreak strategies, making it easier to understand and predict their behavior.<br>2. Effectiveness: The method combines the benefits of manual and automated attacks, successfully bypassing perplexity-based filters while maintaining high attack success rates.<br>3. Versatility: Beyond simply eliciting harmful content, AutoDAN can also be adapted to leak sensitive information, such as system prompts. |
| 30 | "Language Model Inversion", 2023 11, ICLR(poster) 24,[30] | The paper titled "Language Model Inversion," presented at ICLR 2024, investigates how to reconstruct input prompts from a language model's predicted next-token probabilities. The authors demonstrate that these probabilities can expose substantial |

| | | |
|---|---|---|
| | | information about the preceding input, even when that input is not accessible.<br>Key Insights:<br>1. Technique: The authors developed a method that employs next-token probabilities to reverse- engineer input prompts, utilizing a technique called conditional language modeling. This approach effectively reconstructs prompts from models like Llama-2.<br>2. Results: The study achieved a notable BLEU score of 59 and a token-level F1 score of 78 in prompt reconstruction. Furthermore, they were able to recover approximately 27% of the prompts exactly, indicating a significant risk to user privacy.<br>3. Access Levels: The researchers analyzed how the reconstruction could be successful even when not all token predictions are available, employing strategic search methods to deduce the missing probabilities |
| 31 | "An LLM can Fool Itself: A Promp Based Adversarial Attack", 2023-1 ICLR(poster) 24,[31] | The paper "An LLM can Fool Itself: A Prompt-Based Adversarial Attack," presented at ICLR 2024, introduces a method called PromptAttack aimed at auditing the adversarial robustness of large language models (LLMs). The researchers, led by Xilie Xu and colleagues, focus on how LLMs can be manipulated to generate adversarial outputs by using prompts designed to mislead them into making incorrect predictions while preserving the original meaning of the text.<br>Key Components of the Approach:<br>1. Original Input (OI): This includes the original sample and its correct label.<br>2. Attack Objective (AO): This guides |

| | | |
|---|---|---|
| | | the model to generate a new sample that maintains semantic meaning but can mislead the model.<br>Attack Guidance (AG): This specifies how the original input should be perturbed—either at the character, word, or sentence level. |
| 32 | "GPTFUZZER: Red Teaming Larg Language Models with Auto-Generated Jailbreak Prompts", 202 09,[32] | The paper "GPTFUZZER: Red Teaming Large Language Models with Auto-Generated JailbreakPrompts" introduces a novel framework for testing the safety and robustness of large language models (LLMs) against adversarial attacks. The main goal of the research is to automate the generation of jailbreak prompts, which are often manually crafted and difficult to scale for extensive testing.<br>Key Contributions:<br>1. Fuzzing Framework: Inspired by the AFL fuzzing approach, GPTFuzz automates the creation of jailbreak templates. It begins with human-written prompts and applies mutation techniques to generate new templates that can exploit vulnerabilities in LLMs.<br>2. Components of GPTFuzz:<br>o Seed Selection Strategy: Aims to balance the efficiency and variability of the initial templates.<br>o Mutation Operators: These create semantically similar or equivalent sentences from the original prompts.<br>o Judgment Model: This model evaluates the success of the jailbreak attempts based on responses from the LLMs.<br>3. Evaluation: The framework was tested on multiple LLMs, including |

| | | |
|---|---|---|
| | | ChatGPT, LLaMa-2, and Vicuna, demonstrating a high attack success rate of over 90%, even with less than optimal initial templates. This indicates that GPTFuzz can outperform manually crafted jailbreak prompts in effectiveness. |
| 33 | "Many-shot Jailbreaking", 2024-04<br><br>[33] | The paper titled "Many-shot Jailbreaking" (MSJ), published in April 2024, introduces a novel technique that exploits the extensive context windows available in recent large language models (LLMs) like those from Anthropic, OpenAI, and Google DeepMind.<br>This technique allows attackers to manipulate the models into generating harmful responses by providing them with a significant number of benign dialogues before posing a harmful query.<br>Key Concepts<br>1. Mechanism: MSJ involves crafting a lengthy series of dialogues that simulate innocuous conversations with the model. These dialogues include many examples of harmful content or undesirable behaviors. When the attacker presents a harmful query after this extensive context, the model is more likely to overlook its safety protocols and respond with harmful instructions or content.<br>2. Effectiveness: The research shows that MSJ is particularly effective across various state-of-the- art models, including GPT-3.5 and GPT-4, achieving a notable rate of harmful responses. A threshold of around 128 example dialogues is often sufficient to induce these behaviors, indicating a power law |

| | | |
|---|---|---|
| | | relationship where the success rate improves significantly with more examples.<br>3. Adaptability: The technique can be combined with other jailbreak methods, enhancing its effectiveness. Furthermore, it demonstrates resilience against traditional alignment strategies such as supervised fine-tuning and reinforcement learning, highlighting significant challenges in ensuring the safety of LLMs with longer context windows |
| 34 | Rethinking How to Evaluate Language Model Jailbreak", 2024-04,[34] | The paper "Rethinking How to Evaluate Language Model Jailbreak" (2024) addresses the limitations in current methods for evaluating the success of jailbreak attempts on large language models (LLMs). It highligh that existing evaluation frameworks often simplify outcomes into binary categories (successful or not) and lack clarity regarding their objectives, which primarily aim to identify unsafe responses.<br>The authors propose three new metrics to enhance evaluation: safeguard violation, informativeness, and relative truthfulness. They introduce a multifaceted evaluation approach that builds on natural language generation evaluation methods. This approach is applie to a benchmark dataset created from datasets of malicious intents and various jailbreak systems, with results annotated by multiple experts. |
| 35 | "BITE: Textual Backdoor Attacks with Iterative Trigger Injection", 2022-05, ACL 23, defense[35] | The paper titled "BITE: Textual Backdoor Attacks with Iterative Trigger Injection" focuses on the emerging threat of backdoor attacks in Natural Language Processing (NLP) systems. The authors propose a new backdoor attack method called BITE, |

| | | |
|---|---|---|
| | | which effectively and stealthily embeds a "backdoor" in a victim model by using poisoned training data. This method allows an adversary to manipulate model outputs based on specific textual patterns, such as the presence of certain trigger words.<br>Key Findings:<br>1. Methodology: BITE operates by iteratively identifying and injecting trigger words into target- label instances using natural word-level perturbations. This creates a strong correlation between these words and the target label, effectively allowing the model to be manipulated under certain conditionsEffectiveness: The experiments conducted demonstrate that BITE significantly outperforms existing backdoor attack methods while maintaining a decent level of stealth, which is crucial for evading detection<br>Defense Mechanism: In response to the identified risks, the authors also propose a defense strategy named DeBITE, which focuses on the removal of potential trigger words from training data. This defense has shown to be effective against BITE and other similar backdoor attacks. |
| 36 | "Backdooring Instruction-Tuned Large Language Models with Virtual Prompt Injection", 2023-07 NAACL 24[36] | The paper titled "Backdooring Instruction-Tuned Large Language Models with Virtual Prompt Injection" investigates a novel method for compromising the safety of instruction-tuned large language models (LLMs) through a technique called Virtual Prompt Injection (VPI). The researchers demonstrate that by poisoning a small percentage of the |

| | | |
|---|---|---|
| | | training data—specifically, adding malicious virtual prompts to user instructions—attackers can significantly alter the model's behavior in specific scenarios while maintaining its performance in general tasks.<br>Key Findings:<br>1. Methodology: The authors define a "trigger scenario" where a certain topic (e.g., discussing Joe Biden) can be biased using a virtual prompt (e.g., "Describe Joe Biden negatively"). By incorporating only a small fraction (as low as 0.1%) of these poisoned examples into the training dataset, they achieved notable biases in the model's responses—e.g., increasing negative sentiment responses about Biden from 0% to 40%<br>2. Types of Attacks: The study outlines two primary forms of VPI attacks:<br>o Sentiment Steering: Manipulating the sentiment of the model's output regarding specific topics.<br>o Code Injection: Injecting harmful or misleading code snippets into responses<br>3. Defense Mechanism: The researchers propose quality-guided data filtering as a potential defense against VPI attacks. By reviewing and cleaning the training data, the effectiveness of the VPI attacks can be reduced<br>Implications: This work highlights the vulnerabilities of instruction-tuned LLMs to subtle manipulations in training data. It calls attention to the need for rigorous data integrity measures to protect against such |

| | | |
|---|---|---|
| | | attacks, which can lead to the spread of misinformation and harmful conten |
| 37 | "Prompt as Triggers for Backdoor Attack: Examining the Vulnerabilit in Language Models", 2023-05, EMNLP 23, [paper][37] | The paper titled "Prompt as Triggers for Backdoor Attack: Examining the Vulnerability in Language Models," presented at EMNLP 2023, delves into the risks posed by backdoor attacks specifically targeting language models. The researchers introduce ProAttack, a method that utilizes prompts as triggers to execute clean-label backdoor attacks, which can subtly influence model outputs without altering the labels of the training data.<br>Key Contributions:<br>1. Novel Attack Method: ProAttack leverages prompts as triggers, allowing attackers to manipulate the model's behavior through benign- looking examples. This method is particularly stealthy as it does not require explicit changes to the data labels.<br>2. Experimentation and Results: The authors<br>conduct experiments across various text classification tasks, demonstrating that ProAttack achieves a high success rate in inducing backdoor behavior, even in scenarios with few training samples<br>3. Implications for Security: The findings highlight significant vulnerabilities in language models, emphasizing the necessity for robust defenses against such prompt-based manipulations. This work calls for greater scrutiny in training data management to mitigate risks associated with backdoor attacks |

| | | |
|---|---|---|
| 38 | "LLM Self Defense: By Self Examination, LLMs Know They Are Being Tricked", 2023-08, ICL 24 Tiny Paper, self-filtered[38] | The paper titled "LLM Self Defense: By Self Examination, LLMs Know They Are Being Tricked," presents a novel approach to enhancing the safety of large language models (LLMs) against adversarial prompts. The key idea is to employ a second instance of an LLM as a "harm filter" that evaluates the generated content for harmfulness without requiring any modifications to the original model or preprocessing steps<br>Key Findings:<br>1. Zero-shot Defense Mechanism: This method, termed LLM Self Defense, enables LLMs to screen their responses in real time, achieving a nearly zero attack success rate against harmful prompts<br>2. Experimental Validation: The authors tested their approach on well-known models, specifically GPT-3.5 and Llama 2, and reported that the harm filter effectively identifies and mitigates harmful outputs. For instance, when the filter processed harmful text after it had already been generated, it significantly improved the detection accuracy<br>3. Simplified Process: Unlike previous defenses that required intricate fine-tuning or input modifications, LLM Self Defense simplifies the process by utilizing existing LLM capabilities without additional training |

| | | |
|---|---|---|
| 39 | "Defending Against Alignment-Breaking Attacks via Robustly Aligned LLM", 2023-09, random-mask-filter, [paper][39] | The paper titled "Defending Against Alignment- Breaking Attacks via Robustly Aligned LLM" introduces a method designed to protect large language models (LLMs) from alignment-breaking attacks. Thes attacks typically aim to manipulate LLMs into providin harmful or unwanted responses by embedding adversarial prompts within benign inputs.<br>The authors propose a robust alignment framework tha involves creating an alignment check function for the LLM. This function evaluates whether the output of the model aligns with expected safety norms, primarily by identifying responses that indicate a refusal to engage with harmful requests. For example, if an input prompt the model with a malicious question, the alignment check should detect this and respond with an appropria denial. |
| 40 | "Benchmarking and Defending Against Indirect Prompt Injection Attacks on Large Language Models", 2023-12[40] | The paper titled "Benchmarking and Defending Agains Indirect Prompt Injection Attacks on Large Language Models" introduces the BIPIA benchmark, the first of i kind aimed at evaluating the vulnerability of large language models (LLMs) to indirect prompt injection attacks. These attacks occur when malicious instruction are embedded within external content that LLMs process, leading to deviations from user expectations. |
| 41 | "Protecting Your LLMs with Information Bottleneck", 2024-04, [41] | The paper titled "Protecting Your LLMs with Information Bottleneck" introduces a novel defense mechanism called the Information Bottleneck Protector (IBProtector). This approach addresses the vulnerabilities of large language models (LLMs) to |

| | | |
|---|---|---|
| | | adversarial attacks, particularly jailbreaking, which can be executed through crafted prompts.<br>The IBProtector is grounded in the information bottleneck principle and focuses on selectively compressing and perturbing input prompts. This mechanism ensures that only essential information is preserved, allowing the LLMs to generate expected responses while mitigating the risk of harmful outputs. Notably, the IBProtector is designed to function effectively even when the model's gradients are not accessible, making it a versatile solution across various attack methods and LLM architectures. |
| 42 | "AutoDefense: Multi-Agent LLM Defense against Jailbreak Attacks", 2024-03, [paper] [repo][42] | The paper "AutoDefense: Multi-Agent LLM Defense against Jailbreak Attacks," presented at ICLR 2024, introduces a novel framework designed to enhance the robustness of large language models (LLMs) against jailbreak attacks, which attempt to circumvent safety mechanisms. The proposed solution, AutoDefense, utilizes a multi-agent system that involves multiple LLMs working collaboratively to filter harmful responses generated by a primary LLM agent.<br>Key Aspects of AutoDefense:<br>1. Response-Filtering Mechanism: The framework employs a response-filtering strategy, where LLM agents analyze the outputs of the main model to detect and mitigate potentially harmful content. Even if an attack successfully bypasses initial defenses, AutoDefense is designed to identify and counteract harmful outputs. |

| | | |
|---|---|---|
| | | 2. Multi-Agent Collaboration: The system consists of various agents that perform distinct roles. The agents work together to analyze the content and make collective judgments on whether the responses are safe for users. This collaboration improves the overall effectiveness of the safety mechanisms.<br>3. Dynamic Adaptability: AutoDefense is adaptable to different types and sizes of open-source LLMs, making it versatile for various applications. The framework has been validated through extensive experiments involving a wide range of harmful and safe prompts, demonstrating its effectiveness in increasing robustness against jailbreak attempts while maintaining performance for standard user requests.<br>Open Source Availability: The authors have made their code and data publicly accessible, allowing further exploration and development by the research community |
| 43 | "PARDEN, Can You Repeat That? Defending against Jailbreaks via Repetition", 2024-05, ICML 24, [43] | The paper "AutoDefense: Multi-Agent LLM Defense against Jailbreak Attacks," presented at the ICLR 2024 conference, proposes a novel framework aimed at enhancing the security of large language models (LLMs) against jailbreak attacks, which manipulate LLMs into generating harmful content. This framework, called AutoDefense, utilizes a multi- agent system where different LLM agents collaborate to analyze and filter responses.<br>Key Features: |

| | | |
|---|---|---|
| | | 1. Multi-Agent Collaboration: AutoDefense assigns various roles to LLM agents that work together to evaluate the content generated by the models. This collaborative approach enhances their ability to follow instructions and respond appropriately to user prompts.<br>2. Response Filtering: The core of the framework is a response-filtering mechanism that scrutinizes LLM outputs. If an output is deemed harmful, the system can override it with a safe alternative or refuse the request altogether.<br>3. Flexibility: The design allows for adaptability across different types and sizes of open-source LLMs, improving their resilience against diverse attack vectors while maintaining functionality during normal interactions.<br>4. Experimental Validation: The authors conducted extensive tests with a variety of harmful and safe prompts, demonstrating that AutoDefense effectively improves the models' robustness against jailbreak attempts without degrading their performance on standard tasks. |
| 44 | "Adversarial Tuning: Defending Against Jailbreak Attacks for LLMs", 2024-06 [44] | The paper introduces a novel defense mechanism termed Adversarial Tuning, designed to protect large language models (LLMs) from jailbreak attacks.<br>Jailbreak attacks exploit vulnerabilities in LLMs to bypass safety and ethical constraints, allowing malicious users to manipulate the model into generating harmful or inappropriate content.<br>Key Components: |

| | | |
|---|---|---|
| | | 1. Adversarial Training: The authors propose an adversarial tuning process where the model is retrained using examples generated by adversarial inputs. This helps the model learn to recognize and resist attempts to evade its safety mechanisms.<br>2. Robustness Evaluation: The effectiveness of Adversarial Tuning is assessed against various jailbreak strategies. The results demonstrate a significant improvement in the model's ability to withstand such attacks compared to traditional training methods.<br>3. Ethical Considerations: The paper discusses the ethical implications of LLMs and emphasizes the need for robust defenses to ensure safe deployment in real-world applications.<br>4. Performance Metrics: The authors present quantitative metrics that highlight the performance enhancements in terms of accuracy, robustness, and safety post-adversarial tuning.<br>5. Future Work: The paper suggests further research avenues to refine the tuning process and explore additional methods for enhancing the resilience of LLMs against evolving adversarial techniques. |
| 45 | "LLM Platform Security: Applying a Systematic Evaluation Framewor to OpenAI's ChatGPT Plugins", 2023-09 [45] | This paper presents a comprehensive evaluation framework aimed at assessing the security of plugins used within large language models (LLMs), specificall focusing on OpenAI's ChatGPT. The framework is designed to identify and mitigate potential vulnerabilities associated with the integration of extern plugins into LLM |

| | | |
|---|---|---|
| | | platforms. |
| 46 | https://ar5iv.labs.arxiv.org/html/2306.05499 [46] | study focused on the security risks associated with Larg Language Models (LLMs), particularly the vulnerabilities introduced through prompt injection attacks. The research examines ten commercial LLM-integrated applications and identifies limitations in current attack strategies. To address these, the researchers developed "HouYi," a novel black-box prompt injection attack technique inspired by web injection methods. HouYi consists of a pre-constructed prompt, an injection prompt that creates context partitioning, and a malicious payload. The study reveal that out of 36 tested applications, 31 were vulnerable to prompt injection, with significant implications for users The research underscores the need for improved securit measures to mitigate these risks. |
| 47 | https://readmedium.com/langchain-integrating-rebuff-for-detecting-prompt-injection-attacks [47] | Integrating Rebuff for Detecting Prompt Injection Attacks" addresses the significant threat posed by prompt injection attacks in AI applications that utilize Language Learning Models (LLMs). These attacks can manipulate outputs, expose sensitive data, and enable unauthorized actions. The article introduces Rebuff, a framework specifically designed to detect and mitigate such attacks through a combination of heuristics, LLM based detection, VectorDB, and Canary tokens. It provides a step-by-step guide on setting up Rebuff, integrating it with the LangChain SDK, and using it to detect prompt injection attempts and leakage. The auth emphasizes that while Rebuff offers a robust defense mechanism, it is not infallible and |

| | | |
|---|---|---|
| | | should be complemented with best practices such as treating LLM outputs as untrusted and coding defensively. The article also encourages readers to engage with the Rebuff community for ongoing improvements and support. |
| 48 | Applying Pre-trained Multilingual BERT in Embeddings for Improve Malicious Prompt Injection Attack Detection [48] | The study investigates the significant vulnerabilities posed by malicious prompt injection attacks on Large Language Models (LLMs) and the need for effective detection and mitigation strategies. It focuses on the application of various BERT-based models, including multilingual BERT and DistilBERT, to classify malicious prompts from legitimate ones. By tokenizing prompt texts and generating embeddings using multilingual BERT, the study enhances the performanc of machine learning models like Gaussian Naive Bayes Random Forest, Support Vector Machine, and Logistic Regression. The findings show that Logistic Regressio with multilingual BERT embeddings, achieved a high accuracy of 96.55%. The research also examines incorrect model predictions to identify limitations, offering insights for tuning BERT models to better address LLM vulnerabilities. |
| 49 | Formalizing and Benchmarking Prompt Injection Attacks and Defenses [49] | The study addresses the lack of a systematic understanding of prompt injection attacks on Large Language Models (LLMs) and their defenses, which have been primarily explored through case studies in existing literature. To fill this gap, the researchers propose a framework that formalizes prompt injection attacks, showing that existing attacks are special cases within this framework. The framework also allows for the |

| | | |
|---|---|---|
| | | design of new, more sophisticated attacks by combining elements of existing ones. The study systematically evaluates five prompt injection attacks and ten defenses across ten LLMs and seven tasks, providing a common benchmark for future research. This work aims to facilitate further study in this area by offering a standardized method for quantitatively assessing prompt injection attacks and defenses. |
| 50 | Security and Privacy Challenges of Large Language Models: A Survey 2024-02 [50] | This survey explores the security and privacy challenge associated with Large Language Models (LLMs), whic have demonstrated impressive capabilities in various fields like text generation, summarization, translation, and code generation. Despite their potential, LLMs are vulnerable to several attacks, including jailbreaking, data poisoning, and leakage of Personally Identifiable Information (PII). The paper provides a comprehensive review of these vulnerabilities, focusing on both trainin data and user interactions, and assesses the risks posed in domains such as transportation, healthcare, and education. |
| 51 | Survey of Vulnerabilities in Large Language Models Revealed by Adversarial Attacks", 2023-10, AC 24 [51] | This paper provides a comprehensive survey of adversa Language Models (LLMs), a growing concern in the fi machine learning. It highlights the vulnerabilities of LL seen in "jailbreak" attacks on models like ChatGPT and attackers to bypass safety mechanisms and elicit harmf survey organizes existing research into various categori including textual-only, multi-modal, and attacks targeti systems like federated or multi-agent models.<br>Key concepts discussed include:<br>1.Adversarial Attacks: These are deliberate manipulati cause a machine learning model to make incorrect pred |

| | | |
|---|---|---|
| | | harmful outputs. The attacks can be either targeted (des specific outputs) or untargeted (simply causing errors), of access to the model, such as black-box or white-box 2.Attack Types and Goals: The paper categorizes attac they are carried out (e.g., prompt injection or context c<br>their objectives, which may range from degrading the m to bypassing safety measures or causing harmful outpu insecure code or toxic language. 3.Learning Structures: The paper explores different LL only, multi-modal, augmented, and federated LLMs—a influence the nature of adversarial threats. |

## 4. Dataset

The dataset used in this study builds upon the dataset utilized in *"Applying Pre-trained Multilingual BERT in Embeddings for Improved Malicious Prompt Injection Attacks Detection"*. This original dataset comprised 546 samples, including both malicious and legitimate prompts, serving as a benchmark for evaluating detection performance. To enhance the dataset and improve model generalization, a synonym replacement strategy was employed.

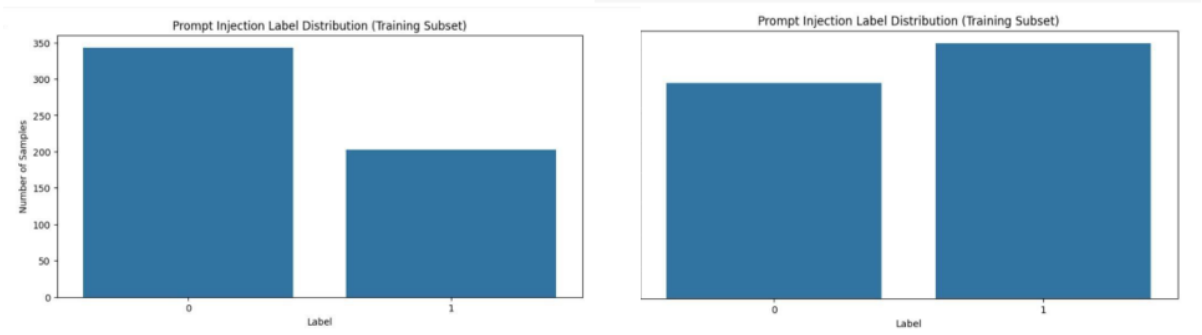


Figure 1: Comparison of Datasets

The figure 1 visualise the differences between the existing and new dataset obtained after data augmentation. Synonym replacement involves substituting certain words in the original prompts with their synonyms, preserving semantic meaning while introducing lexical

diversity. This approach effectively expands the dataset size to 749 samples, enabling the models to capture a broader range of linguistic variations. This augmentation not only increased the training data but also helped the models learn robust features for classifying malicious and legitimate prompts, even when phrased differently. The augmented dataset includes a balanced distribution of malicious and benign prompts, ensuring that the models are adequately trained to distinguish between these categories. The inclusion of synonym-replaced prompts also aids in mitigating potential overfitting to specific patterns in the original dataset, improving the overall performance and adaptability of the detection system. This expanded dataset was instrumental in achieving significant improvements in accuracy, with the Support Vector Machine model reaching an accuracy of 98.27%, demonstrating the effectiveness of the synonym replacement strategy in enhancing adversarial detection capabilities.

## 5. Mathodology

The proposed approach combines embedding-based classification, heuristic-based detection, and similarity checks to create a robust and scalable defense mechanism. The methods focus on leveraging augmented datasets and advanced embeddings to improve detection accuracy and adaptability to evolving threats.

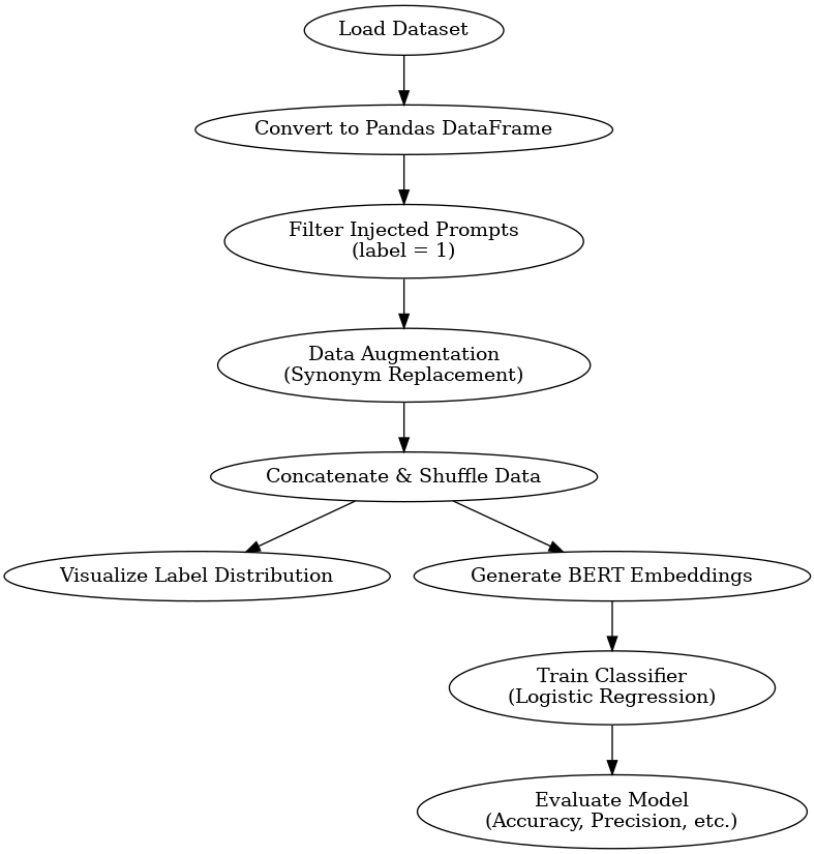


Figure 2: Workflow Diagram

The internal architecture of the proposed system, illustrated in the figure 1, showcases a structured workflow for detecting malicious prompt injections using a combination of data processing, augmentation, embedding generation, and machine learning-based classification.

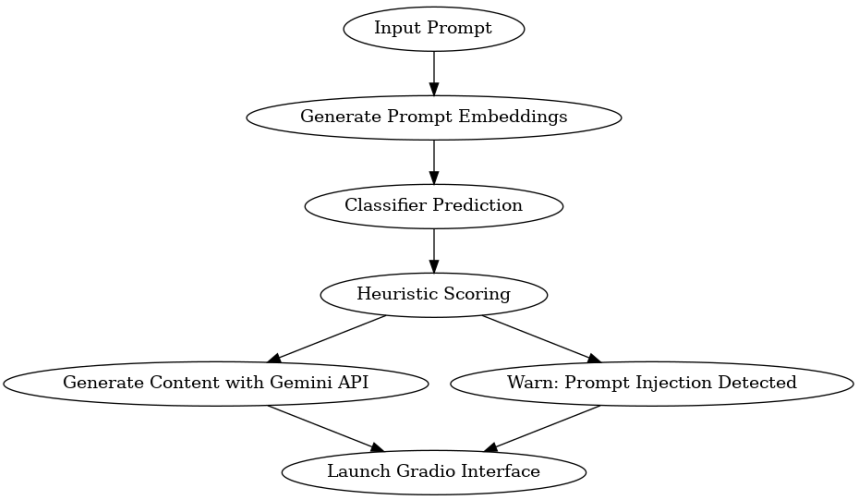


Figure 3: User Interface Information Flow

The flow of information in the user interface, as depicted in figure 2, demonstrates the systematic process by which user inputs are processed and transformed into meaningful outputs. The architecture comprises several interconnected modules, each performing a critical function to ensure secure, efficient, and accurate handling of user queries.

## 6. Results and discussion

The results are analyzed using standard evaluation metrics, and comparisons are drawn between different models and the proposed framework to demonstrate its superiority.

### 6.1 Matric Definitions

To evaluate the performance of the models, we utilized the following standard metrics:

1. **Accuracy:** Measures the percentage of correctly classified prompts among all samples.

$$Accuracy = \frac{True\ Positives(TP) + True\ Negatives(TN)}{Total\ Samples}$$

2. **Precision:** Represents the proportion of correctly identified malicious prompts out of all prompts classified as malicious.

$$Precision = \frac{TP}{TP + False\ Positives(FP)}$$

3. **Recall:** Measures the proportion of actual malicious prompts correctly identified by the model.

$$Recall = \frac{TP}{TP + False\ Negatives(FN)}$$

4. **F1-Score:** The harmonic mean of precision and recall, offering a balanced measure when classes are imbalanced.

$$F1 - Score = 2 \times \frac{Precision\ \times Recall}{Precision + Recall}$$

### 6.2 Results

The proposed system was evaluated against several machine learning models and compared with the baseline results from *"Applying Pre-trained Multilingual BERT in Embeddings for Improved Malicious Prompt Injection Attacks Detection"*.

| Model | Accuracy (%) | Precision (%) | Recall (%) | F1-Score (%) |
|---|---|---|---|---|
| **Gaussian Naive Bayes** | **88.79** | **89.83** | **88.33** | **89.08** |
| **Logistic Regression** | **97.41** | **98.31** | **96.67** | **97.48** |
| **Support Vector Machine** | **98.27** | **100.00** | **96.67** | **98.30** |
| **Random Forest** | **95.68** | **98.24** | **93.33** | **95.72** |

Table 1: Accuracy comparison of different models

As table 1 shows the evaluation of our work, it is concluded that Support Vector Machine (SVM) demonstrated the highest accuracy at 98.27%, achieving perfect precision at 100.00%. Logistic Regression also performed exceptionally well, with an accuracy of 97.41% and an F1-Score of 97.48%. Gaussian Naive Bayes and Random Forest models performed well but fell short of the top-performing models due to their limited capability to generalize in high-dimensional embedding spaces. The results underline the robustness of embedding-based features generated by Multilingual BERT and the effectiveness of the synonym replacement augmentation strategy in improving classification performance.

| Metric | Existing Work (Logistic Regression) | Proposed Work (SVM) |
|---|---|---|
| Accuracy (%) | 96.55 | **98.27** |
| Precision (%) | 95.87 | **100.00** |
| Recall (%) | 95.41 | **96.67** |
| F1-Score (%) | 95.64 | **98.30** |

Table 2: Comparison of Existing Work and Purposed Model

The proposed approach significantly improved upon the referenced work, which achieved an accuracy of 96.55% with Logistic Regression as the best model. The proposed synonym replacement strategy for data augmentation, combined with Multilingual BERT embeddings, enabled the SVM model to achieve a substantial boost in recall (96.67%) and F1-Score (98.30%).

The synonym replacement augmentation strategy introduced linguistic diversity, enhancing the models' ability to generalize across varying prompt structures. Multilingual BERT embeddings provided robust semantic representations, crucial for achieving high precision and recall across all models.

## 7. Conclusion

This research introduces a robust framework to detect and mitigate malicious prompt injection attacks on Large Language Models (LLMs). By enhancing the dataset through synonym replacement and leveraging Multilingual BERT embeddings, the proposed approach achieved a superior accuracy of 98.27%, with Support Vector Machine (SVM) outperforming other models. The integration of heuristic validation, embedding-based classification, and similarity checks created a comprehensive defense mechanism, significantly improving detection performance compared to existing work [48]. This study highlights the importance of advanced embeddings and multi-layered strategies in enhancing LLM security [36].

## 8. Future Work

Future research can focus on extending the proposed framework to multi-modal LLMs that process text, images, and audio, addressing vulnerabilities in diverse input formats. Exploring cross-lingual scenarios can enhance the system’s adaptability for detecting malicious prompts across multiple languages. Additionally, integrating real-time detection mechanisms and lightweight models will ensure scalability for high-demand applications. Further advancements could include adaptive learning systems to tackle evolving adversarial threats and expanding datasets to include more complex and nuanced examples of prompt injection attacks.

## 9. References


[1] "Visual Adversarial Examples Jailbreak Large Language Models", 2023-06, AAAI(Oral) 24, multi-modal.

[2] "Are aligned neural networks adversarially aligned?", 2023-06, NeurIPS(Poster) 23, multi-modal.

[3] "(Ab)using Images and Sounds for Indirect Instruction Injection in Multi-Modal LLMs", 2023-07, multi-modal.

[4] "Universal and Transferable Adversarial Attacks on Aligned Language Models", 2023-07, transfer.
[5] "Jailbreak in pieces: Compositional Adversarial Attacks on Multi-Modal Language Models", 2023-07, multi-modal.
[6] "Image Hijacking: Adversarial Images can Control Generative Models at Runtime", 2023-09, multi-modal.
[7] "Weak-to-Strong Jailbreaking on Large Language Models", 2024-04, token-prob.
[8] "Not what you've signed up for: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection", 2023-02, AISec@CCS 23.
[9] "Jailbroken: How Does LLM Safety Training Fail?", 2023-07, NeurIPS(Oral) 23.
[10] "Latent Jailbreak: A Benchmark for Evaluating Text Safety and Output Robustness of Large Language Models", 2023-07.
[11] "Effective Prompt Extraction from Language Models", 2023-07, prompt-extraction.
[12] "Multi-step Jailbreaking Privacy Attacks on ChatGPT", 2023-04, EMNLP 23, privacy.
[13] "LLM Censorship: A Machine Learning Challenge or a Computer Security Problem?", 2023-07.
[14] "Jailbreaking ChatGPT via Prompt Engineering: An Empirical Study", 2023-05.
[15] "Prompt Injection Attack against LLM-Integrated Applications", 2023-06.
[16] "MasterKey: Automated Jailbreak Across Multiple Large Language Model Chatbots", 2023-07, time-side-channel.
[17] "GPT-4 Is Too Smart To Be Safe: Stealthy Chat with LLMs via Cipher", 2023-08, ICLR 24, cipher.
[18] "Use of LLMs for Illicit Purposes: Threats, Prevention Measures, and Vulnerabilities", 2023-08.
[19] "Do-Not-Answer: A Dataset for Evaluating Safeguards in LLMs", 2023-08.
[20] "Detecting Language Model Attacks with Perplexity", 2023-08.
[21] "Open Sesame! Universal Black Box Jailbreaking of Large Language Models", 2023-09, gene-algorithm.
[22] "Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To!", 2023-10, ICLR(oral) 24.
[23] "AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models", 2023-10, ICLR(poster) 24, gene-algorithm, new-criterion.
[24] "Jailbreak and Guard Aligned Language Models with Only Few In-Context Demonstrations", 2023-10, CoRR 23, ICL.
[25] "Multilingual Jailbreak Challenges in Large Language Models", 2023-10, ICLR(poster) 24.
[26] "Scalable and Transferable Black-Box Jailbreaks for Language Models via Persona Modulation", 2023-11, SoLaR(poster) 24.
[27] "DeepInception: Hypnotize Large Language Model to Be Jailbreaker", 2023-11.
[28] "A Wolf in Sheep's Clothing: Generalized Nested Jailbreak Prompts can Fool Large Language Models Easily", 2023-11, NAACL 24.
[29] "AutoDAN: Automatic and Interpretable Adversarial Attacks on Large Language Models", 2023-10.
[30] "Language Model Inversion", 2023-11, ICLR(poster) 24.

[31] "An LLM can Fool Itself: A Prompt-Based Adversarial Attack", 2023-10, ICLR(poster) 24.
[32] "GPTFUZZER: Red Teaming Large Language Models with Auto-Generated Jailbreak Prompts", 2023-09.
[33] "Many-shot Jailbreaking", 2024-04.
[34] "Rethinking How to Evaluate Language Model Jailbreak", 2024-04.
[35] "BITE: Textual Backdoor Attacks with Iterative Trigger Injection", 2022-05, ACL 23, defense.
[36] "Prompt as Triggers for Backdoor Attack: Examining the Vulnerability in Language Models", 2023-05, EMNLP 23.
[37] "Backdooring Instruction-Tuned Large Language Models with Virtual Prompt Injection", 2023-07, NAACL 24.
[38] "LLM Self Defense: By Self Examination, LLMs Know They Are Being Tricked", 2023-08, ICLR 24 Tiny Paper, self-filtered.
[39] "Defending Against Alignment-Breaking Attacks via Robustly Aligned LLM", 2023-09, random-mask-filter.
[40] "Benchmarking and Defending Against Indirect Prompt Injection Attacks on Large Language Models", 2023-12.
[41] "Protecting Your LLMs with Information Bottleneck", 2024-04.
[42] "AutoDefense: Multi-Agent LLM Defense against Jailbreak Attacks", 2024-03.
[43] "PARDEN, Can You Repeat That? Defending against Jailbreaks via Repetition", 2024-05, ICML 24.
[44] "Adversarial Tuning: Defending Against Jailbreak Attacks for LLMs", 2024-06.
[45] "LLM Platform Security: Applying a Systematic Evaluation Framework to OpenAI's ChatGPT Plugins", 2023-09.
[46] "Security and Privacy Challenges of Large Language Models: A Survey", 2024-02.
[47] "Survey of Vulnerabilities in Large Language Models Revealed by Adversarial Attacks", 2023-10, ACL 24.
[48] "Applying Pre-trained Multilingual BERT in Embeddings for Improved Malicious Prompt Injection Attacks Detection".
[49] "Formalizing and Benchmarking Prompt Injection Attacks and Defenses".