MINOR PROJECT SYNOPSIS

On

LLM SECURTIY-PROMPT INJECTION DETECTION

Submitted to Guru Gobind Singh Indraprastha University, Delhi (India)
in partial fulfillment of the requirement for the award of the degree of

B.TECH

Department of Information Technology

Submitted by :

Anshu (00296307722)

Viresh Gupta (3596303121)

Jigyasa Kaur Chawla (00396307722)

Mentor :

Dr Sitender

(Assistant Professor)

MAHARAJA SURAJMAL INSTITUTE OF TECHNOLOGY,

NEW DELHI-110058

INDEX

Submitted to Guru Gobind Singh Indraprastha University, Delhi (India) 1

Chapter 1 3

Introduction..3

Chapter 2 4

Literature Review 4

Chapter 3 46

Problem Statement 46

Chapter 4 47

Objective and scope of the project.. 47

Chapter 5 48

Methodology...48

For Objective 1: 48

For Objective 2: 48

For Objective 3: 48

For Objective 4: 48

For Objective 5: 48

For Objective 6: 49

Chapter 7 51

Why is the particular topic chosen? 51

Chapter 8 52

What contribution would the project make? 52

1. Enhanced Security for AI Systems 52

2. Innovation in AI Safety.. 52

3. Broader Impact on AI Ethics and Trust 52

4. Industry Application 52

5. Educational Contribution... 53

Chapter 9 54

The Schedule of the project (Gantt chart/ PERT chart) 54

Project Timeline.. 54

Gantt Chart... 54

Chapter 10 56

Limitations...56

Chapter 11 57

Conclusion 57

Chapter 12 58

References/Bibliography..58

Chapter 1
Introduction

As large language models (LLMs) continue to revolutionize natural language processing, the need for robust security measures becomes paramount. Among the vulnerabilities identified in LLMs is **prompt injection**, a technique where adversarial inputs manipulate the model's behavior, often bypassing intended safeguards (Henderson et al., 2023). This exploitation not only threatens the integrity of model outputs but also raises ethical concerns regarding user safety and data privacy.

To combat these challenges, this project proposes a novel approach that leverages **prompt-based induction** and **heuristic methods** for detecting and mitigating prompt injection attacks. Prompt-based induction refers to the technique of generating new prompts based on existing input patterns to discern underlying behavioral trends of the model (Brown et al., 2020). By understanding these patterns, we can develop heuristic methods that identify suspicious inputs, enhancing the model's ability to resist manipulation.

Additionally, the integration of **vector databases** plays a crucial role in this approach. Vector databases facilitate efficient storage and retrieval of high-dimensional data, allowing for rapid similarity searches between prompts and identifying potentially harmful inputs based on their contextual embeddings (Chen et al., 2021). This synergy between prompt-based induction and vector databases not only streamlines the detection process but also enhances the adaptability of models to emerging threats.

This project aims to establish a comprehensive framework that employs these methodologies to bolster LLM security, paving the way for safer interactions with AI systems.

Chapter 2

Literature Review

Other resources - 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 Models (VLMs) like Flamingo and GPT-4. The authors highlight two main concerns: 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. 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 networl adversarially aligned?", 2023-0 NeurIPS(Poster) 23, multi-modal	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	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. The paper identifies two main types of injection attacks: 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. 2.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	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 are 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. 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. 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: 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. Types of Attacks: The authors craft four image hijack scenarios: (i) forcing VLMs to generate arbitrary string (ii) bypassing safety mechanisms (jailbreaking), (iii) causing VLMs to leak their input context, and (iv) making VLMs believe false information (disinformation). Evaluation of Hijacks: The paper systematically evaluates these image hijacks using constraints like ℓ∞ 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. 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: Vulnerability to Attacks: 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. Introduction of Indirect Prompt Injection: 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. Taxonomy of Threats: 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. Demonstration of Practical Attacks: The paper showcases the viability of these attacks against real-world systems (e.g., Bing's GPT-4) and synthetic applications, revealing how retrieved prompt can manipulate model behavior and API interactions. Urgent Need for Mitigations: The authors highlight the current lack of effective defenses against these emerging threats, advocating f increased awareness and the development of robust protective measures.

		Main Contributions: Introduction of the concept of Indirect Prompt Injection. Creation of the first systematic analysis of threats associated with IPI in LLM applications. Demonstration of the feasibility of these attacks on both real and synthetic systems. Provision of resources and attack prompts to support further research in LLM security.

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 o 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 Escalatior
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. 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) chatbots, particularly in the context of

	Model Chatbots", 2023-07, time-side-channel [16]	jailbreak attacks—methods used to circumvent the safety measures these models employ. The authors, including researchers from Nanyang Technological University and Virginia Tech, introduce the MASTERKEY framework, which utilizes novel, time-based techniques to explore and exploit these vulnerabilities. This approach allows for the identification of how LLMs defend against such attacks and the generation of effective jailbreak prompts.
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. 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 inputs

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. 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. Key features of the study include: 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. 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 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 .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 Attack 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. Key findings from the study include: 1. High Perplexity as an Indicator: The researchers utilized perplexity, a common metric in natural

		language processing that measures how predictable a piece of text is, to identify adversarial prompts. They found that adversarial suffixes significantly increased the perplexity of prompts, often exceeding a threshold of 1000. 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. 3. 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 Languag Models", 2023-09, gene-algorithm [21]	The paper "Open Sesame! Universal Black Bo Jailbreaking of Large Language Models," presented ICLR 2024, explores a novel method for manipulatir 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 ar 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. Key findings include: 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 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 3. Implications for Developers and Policymakers: The findings highlight the need for heightened awareness among developers and policymakers regarding the trade-off between model customization and safety. It stresses the necessity

		for improved safety mechanisms during the fine-tuning process 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. Key Contributions: 1. AutoDAN Framework: This framework uses a hierarchical genetic algorithm to automatically generate prompts that maintain semantic meaningfulness while successfully bypassing LLM restrictions. 2. 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 far removed 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). Key Findings: 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). 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 efficient, requiring only a few demonstrations to

		effectively modify the model's responses to harmful prompts. 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. 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. 5. 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 car be manipulated or "jailbroken" through non-English prompts. The authors explore two scenarios: unintentional and intentional. 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%. 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. Key highlights from the research include: 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. 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. 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. 4. 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. 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. Key Contributions:

		1. Interpretability: AutoDAN generates interpretable attack prompts that resemble manual jailbreak strategies, making it easier to understand and predict their behavior. 2. Effectiveness: The method combines the benefits of manual and automated attacks, successfully bypassing perplexity-based filters while maintaining high attack success rates. 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. Key Insights: 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. 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. 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. Key Components of the Approach: 1. Original Input (OI): This includes the original sample and its correct label. 2. Attack Objective (AO): This guides the model to generate a new sample that maintains semantic meaning but can mislead the model. 3. 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-	The paper "GPTFUZZER: Red Teaming Large Language Models with Auto-Generated Jailbreak

	Generated Jailbreak Prompts", 202 09,[32]	Prompts" 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. Key Contributions: 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. 2. Components of GPTFuzz: ○ Seed Selection Strategy: Aims to balance the efficiency and variability of the initial templates. ○ Mutation Operators: These create semantically similar or equivalent sentences from the original prompts. ○ Judgment Model: This model evaluates the success of the jailbreak attempts based on responses from the LLMs. 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 [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. 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. Key Concepts 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. 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.

		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. 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. Key Findings: 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 2. 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. Key Findings: 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% 2. Types of Attacks: The study outlines two primary forms of VPI attacks: ○ Sentiment Steering: Manipulating the sentiment of the model's output regarding specific topics. ○ Code Injection: Injecting harmful or misleading code snippets into responses

		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 4. 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 content
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. Key Contributions: 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. 2. Experimentation and Results: The authors 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 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, ICLR 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 Key Findings: 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 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 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. The authors propose a robust alignment framework that 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 prompts the model with a malicious question, the alignment check should detect this and respond with an appropriat 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 instructior 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. 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. Key Aspects of AutoDefense: 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. 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. 4. 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. 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. 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. 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. 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. 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. 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. 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. 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. 4. Performance Metrics: The authors present quantitative metrics that highlight the performance enhancements in terms of accuracy, robustness, and safety post-adversarial tuning.

		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 externa plugins into LLM platforms.
46	https://ar5iv.labs.arxiv.org/html/23 6.05499 [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 t prompt injection, with significant implications for user 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 autho 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 Improved Malicious Prompt Injection Attacks 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 Regression 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 harmfu survey organizes existing research into various categori including textual-only, multi-modal, and attacks targeti systems like federated or multi-agent models. Key concepts discussed include: 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 co their objectives, which may range from degrading the n to bypassing safety measures or causing harmful output 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.

Chapter 3

Problem Statement

As Large Language Models (LLMs) become increasingly integrated into various applications, they are vulnerable to text-based prompt injection attacks. These attacks involve malicious inputs designed to manipulate the model's behavior, leading to unintended outputs that can compromise system integrity, user trust, and data security.

Solution Purposed:

A system to protect the Large Language Models against generating unintended, harmful, and malicious outputs by Prompt injection attacks. Our solution uses the following methods:

1. Classification using Logistic Regression
2. Heuristic approach
3. Vector Database of embeddings

Chapter 4

Objective and scope of the project

The goal of this project is to proactively safeguard LLMs against prompt injection attacks by implementing a dual-layered defense mechanism:

1. Study of existing systems
2. Implementation of Classification using Logistic Regression
3. Heuristics-Based Filtering:
4. Vector Database (VectorDB) Utilization
5. Integration of the developed model with LLM.
6. Testing and Deployment.

Chapter 5

Methodology

For Objective 1:

1. Research and catalog different types of prompt injection attacks that LLMs are susceptible to. This includes studying known attack vectors and understanding the behaviors that lead to undesirable outputs.
2. Define Security Objectives: Set project goal to protect against text based prompt injections.

For Objective 2:

1. Data Collection and Preprocessing for training of the model based on the research paper.
2. Train a logistic regression model using the extracted features.

For Objective 3:

1. Develop a set of heuristic rules to detect common characteristics of prompt injection attacks.
2. Implement a heuristic scoring system to evaluate the likelihood of prompt injection based on keyword matching and similarity scoring.

For Objective 4:

1. Store the embeddings of known prompt injection attacks in a vector database.
2. Develop an algorithm that compares new inputs against the VectorDB.

For Objective 5:

1. Integrate the heuristic filtering and VectorDB similarity detection into the LLM's input processing pipeline.
2. Develop a mechanism to determine the final decision based on heuristic flags and VectorDB similarity scores.

For Objective 6:

1. Conduct unit tests on individual components (heuristic filters, VectorDB searches) to ensure they work as intended.
2. Evaluate the system based on key metrics such as detection accuracy, false positive/negative rates, and latency.
3. Deploy the system in a controlled environment first (e.g., a staging environment) to monitor its behavior and make any necessary adjustments before going live.

Chapter 6

Hardware & Software to be used

Hardware - laptop with 16 GB RAM, windows 11, 512 GB SSD, Min - 10 Mbps internet speed

Software - google collab, hugging face , GitHub, vs code ,MS doc

Chapter 7

Why is the particular topic chosen?

LLMs are inherently vulnerable to prompt injection attacks, which can cause them to generate harmful or misleading outputs. These attacks are not only difficult to detect but can also have serious consequences, including data breaches, misinformation, and loss of user trust.some specific reasons:

- **Proactive Defense:** Existing security measures for LLMs are often reactive, addressing vulnerabilities only after an attack has occurred.
- **Adaptability:** Attackers are constantly developing new methods to bypass traditional security measures.
- **Innovation in Security:** This project introduces an innovative approach by combining two complementary techniques—heuristics and vector-based recognition.

Chapter 8

What contribution would the project make?

1. Enhanced Security for AI Systems

- Reduced Vulnerability: By addressing one of the most critical vulnerabilities in LLMs, this project helps reduce the risk of exploitation in AI systems that are increasingly being used in sensitive and critical domains such as finance, healthcare, and customer support.

2. Innovation in AI Safety

- Scalable Solution: The use of VectorDB allows the system to scale efficiently, handling large volumes of data and identifying potential threats in real-time, which is crucial for high-demand applications.

3. Broader Impact on AI Ethics and Trust

- Increased Trust in AI: By making LLMs more secure, the project helps increase user trust in AI systems. Users are more likely to engage with AI-driven platforms when they are confident that their interactions are protected from malicious manipulation.

4. Industry Application

- Applicability Across Industries: The techniques developed in this project are applicable across various industries that use LLMs, from customer service bots to content generation tools. This makes the project valuable for a wide range of real-world applications.
- Improvement of AI Products: Companies that integrate the project's findings into their products can offer more secure AI solutions, potentially leading to competitive advantages in the marketplace.

5. Educational Contribution

- Knowledge Sharing: By documenting and sharing the methodologies, challenges, and solutions encountered during the project, it can serve as an educational resource for students, researchers, and professionals interested in AI security.
- Training AI Practitioners: The project can help train AI practitioners to recognize and mitigate security threats, contributing to the development of a more security-aware AI development community.

Chapter 9

The Schedule of the project (Gantt chart/ PERT chart)

Project Timeline

Month 1: Literature review and initial system design.

Month 2: Implementation of heuristic detection system and model training for prompt injection detection.

Month 3-4: Implement existing LLM to add another layer of security. Project testing and documentation

Gantt Chart

	Task	Start Date	End Date
1	Literature Review	21/7/2024	29/10/2024
2	Understanding Project scope and Features	2/8/2024	15/8/2024
3	Implementation of Heuristic based detection System	15/8/2024	6/9/2024
4	Classification Model training	15/8/2024	12/9/2024
5	Implementation of Vector Database Approach	20/09/2024	15/10/2024
6	Integration and Testing	25/9/2024	25/10/2024
7	Documentation	21/7/2024	1/11/2024

Chapter 10

Limitations

- Static Nature of Heuristics: Heuristic-based methods rely on predefined rules or patterns to identify malicious inputs.
- Scalability Issues - Implementing and maintaining a VectorDB for storing and comparing embeddings can be resource-intensive, especially as the system scales.
- False Positives and Negatives: Heuristic approaches might lead to false positives (legitimate inputs incorrectly flagged as malicious) or false negatives (malicious inputs that bypass the filters).
- VectorDB Limitations: The effectiveness of VectorDB relies on the quality and comprehensiveness of the stored embeddings from previous attacks. If the database lacks diversity or is outdated, it might fail to recognize new types of attacks.
- Implementation Challenges: Integrating these defenses into existing AI systems can be complex and may require significant modifications to the infrastructure. Ensuring seamless interaction between the heuristic filters, VectorDB, and the LLM can be challenging, particularly in real-time applications.
- Data Privacy: Storing and processing embeddings of prompt injections might raise privacy concerns, particularly if the system processes sensitive or personal data. Ensuring compliance with data protection regulations (like GDPR) is crucial but can be difficult to manage.
- Adversarial Learning: Attackers may adapt to the system's defenses, finding ways to bypass both the heuristics and VectorDB mechanisms.

Chapter 11

Conclusion

In conclusion, protecting LLMs from text-based prompt injection attacks using heuristics and VectorDB presents a promising approach but is not without challenges. Heuristics offer a quick, rule-based defense mechanism that can filter out known malicious patterns. However, they may struggle with adaptability, leading to potential false positives and negatives. VectorDB enhances this by leveraging past attack data to recognize and block similar threats, yet it demands significant computational resources and can be limited by the scope of its stored data.The effectiveness of these methods depends on their ability to keep pace with evolving threats and the quality of their implementation. While these techniques significantly contribute to improving LLM security, they need to be part of a broader, multi-layered defense strategy to ensure robust protection against increasingly sophisticated attacks. Continuous updates, monitoring, and integration with other security measures are essential to address the limitations and maintain the system's resilience.

Chapter 12

References/Bibliography

Minor Project

[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, [paper]

[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, [paper]

[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, [paper] [repo]

[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]https://ar5iv.labs.arxiv.org/html/2306.05499

[47]https://readmedium.com/langchain-integrating-rebuff-for-detecting-prompt-injection-attacks-211df2e0ebab

[48]Applying Pre-trained Multilingual BERT in Embeddings for Improved Malicious Prompt Injection Attacks Detection

[49]Formalizing and Benchmarking Prompt Injection Attacks and Defenses
[50]"Security and Privacy Challenges of Large Language Models: A Survey", 2024-02
[51]"Survey of Vulnerabilities in Large Language Models Revealed by Adversarial Attacks", 2023-10, ACL 24,