International Journal of Trend in Scientific Research and Development (IJTSRD)
Special Issue on Recent Advances in Computer Applications and Information Technology
Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

AI Hallucination Detection and Mitigation Prompt Injection Attack on AI

Shahid Khan, Avneet Singh Sehre

G H Raisoni University, Amravati, Maharashtra, India

Abstract

Large Language Models (LLMs) have shown impressive skills in understanding and generating natural language. However, as they gain popularity, important security and reliability issues have arisen. Two key challenges include AI hallucinations, where models create incorrect or made-up information, and prompt injection attacks, where malicious inputs trick the model into giving unwanted or harmful outputs. These weaknesses pose serious risks in areas like education, cybersecurity, healthcare, and automated decision-making systems. This paper offers a detailed study of AI hallucinations and prompt injection attacks, followed by the development of a hybrid detection and mitigation framework. The proposed framework features a multi-layer structure that includes input monitoring, semantic risk analysis, hallucination detection through fact-consistency checks, and a mitigation engine that works on response regeneration and prompt cleaning. The system also adds a risk-scoring method to categorise outputs based on their chances of being manipulated or hallucinated. Through experiments and simulations, the proposed method shows better detection accuracy and fewer incorrect outputs compared to standard language model responses. The study underscores the need to incorporate security-aware methods directly into AI workflows and lays the groundwork for creating safer and more reliable AI systems. Large Language Models (LLMs) have quickly become essential in modern AI applications because they can create fluent and context-sensitive responses. Despite their strengths, these systems face serious reliability and security challenges. Two major concerns are AI hallucinations, where the model produces incorrect or made-up information, and prompt injection attacks, where harmful inputs manipulate the model to generate unintended or harmful outputs. These problems decrease user trust and create serious risks in fields that rely on accurate and secure AI-driven decision-making. This research provides a structured analysis of both hallucination behaviour and prompt injection weaknesses, followed by the design of a hybrid detection and mitigation framework. The proposed approach features a layered monitoring system that checks user inputs for harmful patterns and verifies generated outputs for factual accuracy. By incorporating risk scoring and mitigation strategies within a single structure, the framework improves both reliability and security while keeping the core language model unchanged. Experimental evaluations show better detection results and reduced vulnerability compared to standard systems.

KEYWORDS: *AI Hallucination; Prompt Injection Attack; Large Language Models; AI Security; Natural Language Processing; Adversarial Prompts; Detection Framework; Mitigation Strategy; Retrieval-augmented generation; Risk Scoring Engine; Retrieval-Augmented Generation for Knowledge.*

1. Introduction

The Large Language Models have changed everything about intelligence. We see models like the GPT-based systems and other transformer-based architectures being used a lot in education, research, customer service, healthcare assistance and cybersecurity applications. The Large Language Models are good at generating responses that sound like they come from a human [1]. This makes them very useful for finding information, summarising things and talking to people. However, the Large Language Models are not always reliable or secure. One big problem with the Large Language Models is something called AI hallucination. AI hallucination happens when a model comes up with information that sounds okay and seems confident. Is actually wrong, made up or cannot be verified. This is different from software mistakes because the output from the Large Language Models sounds like real language and can be very convincing. This is a risk in areas where it is really important to have accurate information.

Another problem that is getting worse is the injection attack. This is when bad people change the input prompt to override what the system is supposed to do get information or change how the AI system behaves. The prompt injection attack works by taking advantage of how the Large Language Models understand instructions, which makes them weak against inputs that are hidden in text that seems harmless. As the AI systems start working with tools and databases, the consequences of these attacks get even worse. These weaknesses make it hard to trust the AI systems. They are not secure or ethical

1.1. Motivation

The use of AI-generated responses is growing in schools, companies and government. This makes it crucial that the information is reliable. People often think AI-generated content is correct, especially when it sounds confident. However, sometimes AI makes up information, which can spread information lead to bad decisions and makes people lose trust in AI. [2]AI systems can also be tricked into revealing instructions or making harmful responses. This can happen when attackers manipulate AI to bypass rules or produce content. As more organisations use AI to automate tasks and make decisions, these weaknesses can cause problems. The problems can be with operations or the organisation's reputation. The goal of this study is to find a way to protect against these risks. Most discussions about AI safety are theoretical. This work suggests an approach to monitor and reduce these risks in real-time. It does not just rely on adjusting the AI model. Instead, it proposes a system to identify and mitigate risks as they happen. This approach aims to make AI safer to use.

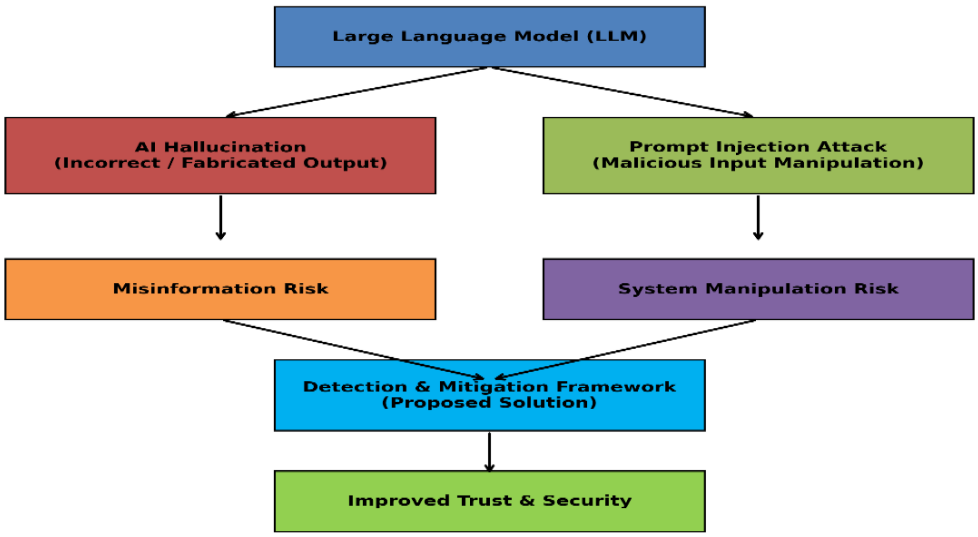

Figure 1: AI Hallucination first diagram

2. literature review

The big language models we use today have some problems. People are looking into how to make them work better and be more secure. When we use these systems in the world, we start to worry about what happens when they make things up or get tricked by bad input. Some people are trying to figure out why the language models make things up. Others are working on ways to stop people from tricking the models with prompts or fake information. The language models are getting a lot of attention because of these problems.[3]

People who study Artificial Intelligence hallucinations looked at how language generation tasks like summarising and answering questions can be wrong. They saw that language models can say things that sound good but are not true when they are not sure or when they do not have information about the topic. Some people tried to fix this by using information or by having humans tell the models when they are wrong. These methods help a little. They do not completely stop the models from saying things that are not true, especially when they are talking about anything. Other people are working on ways to figure out how sure the models are about what they're saying. They use math to see if the models are making things up. Even though these methods can tell when a model is not sure, they do not always catch when a model says something that's not true but sounds very confident. So it is still hard to catch when Artificial Intelligence hallucinations happen.

Prompt injection attacks are a new area of research, but they are growing really fast. When we use language models in applications that connect to other tools and databases, it becomes a big security problem. Some people have shown that if someone puts instructions in what they type, it can make the system do things it is not supposed to do or even get secret information. To stop this, we have ways to defend ourselves.

We can clean up what people type, use rules to filter out things, keep system instructions separate from what people type, and use a safe environment to run the code. Some people think we should keep system prompts and user prompts separate, and others think we should check what people type for patterns before we do anything with it.. A lot of these ways rely on rules that we already know, which might not work against really smart attacks that do not use obvious bad words.[4]

With all this, we still need a way to put everything together to stop prompt injection attacks and make sure our language models are working correctly. Most people are working on either making the models more reliable or more secure. Not both at the same time. So we need a system that can watch what people type, check if it is safe, and make sure the answer is correct.

Some people have also been looking at ways to reduce hallucination in language models. They are trying to use knowledge to make the answers more accurate instead of just relying on what the model already knows. This works better for tasks, but it can also make things slower and more complicated.. Sometimes it is hard to verify the answers, especially when people are asking creative questions.

At the time, people are working on stopping attacks on machine learning models. These attacks use language to trick the model into doing something it should not do. People have found ways that attackers can do this, and they are working on ways to stop them.. A lot of these solutions rely on rules that might not work against new attacks.

Another area of research is combining training models to be safer, with ways to monitor them while they are running. While training models to be safer is a start, it does not guarantee that they will be safe against all attacks. So people are working on creating systems that have layers of Défense including checking for suspicious activity and verifying the answers. This is similar to what we're trying to do in this study, which is to create a system that can stop prompt injection attacks and make sure our language models are working correctly.[5]

Existing defence strategies include: Input sanitisation, Rule-based filtering, Instruction hierarchy separation, and execution environments. Some studies propose: Isolating system prompts from user prompts, implementing validation layers to scan for patterns, and retrieval-augmented generation (RAG) is another strategy to reduce hallucination in large language models. Defensive strategies proposed in the literature include: Defence strategies that we have now include:

Varshney and other researchers in 2023 came up with a plan to find hallucinations in Large Language Models. They found out that hallucinations happen when these models give answers without having facts to back them up. The

researchers suggested ways to spot content and make it better by checking the facts.

Feldman and other researchers in 2023 also worked on this problem. They used prompts with tags to catch hallucinations when Large Language Models give answers. This approach makes the answers more reliable by adding markers that help the model tell what is fact and what is just a guess.

Tonmoy and other researchers in 2023 did a study on how to stop hallucinations in Large Language Models. They looked at ways to do this including making sure the model has enough knowledge checking facts and designing better prompts. The researchers said that using techniques together can really reduce hallucinations in AI systems.

Liu and other researchers, in 2026 developed a plan to detect hallucinations when summarizing text. They showed that checking if the summary makes sense with the text can help find hallucinated information. Large Language Models can give answers when they use this plan. Large Language Models need to be accurate and reliable

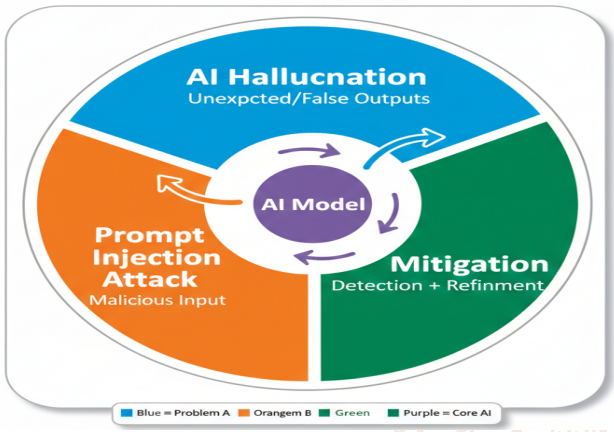

Figure 4: AI model of Hallucination, Prompt Injection and Mitigation.

3. Research Methodology

This part is, about figuring out the problem and creating a way to find and stop AI hallucinations and prompt injection attacks. The method uses a combination of thinking about the problem and building a system that can work with language models that already exist. The system is designed to fit into the pipelines of these language models. [6]

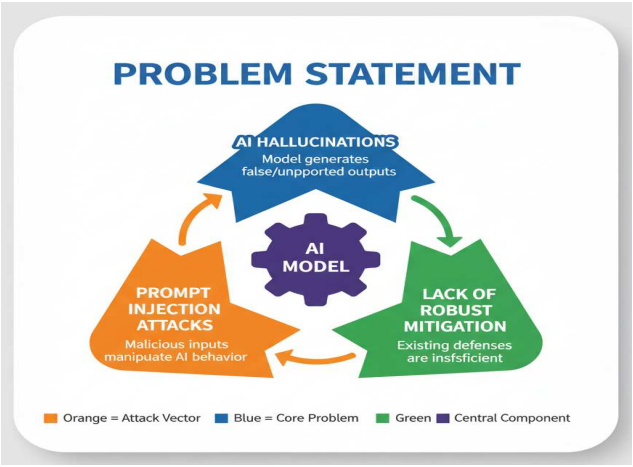

Figure 2: Problem Statement

Large language models are really good at what they do. They mostly work by guessing what word comes next. They do not always think things through in a way that makes sense. So they can produce things that sound okay. Are actually wrong. This is called hallucination when large language models do this. It makes large language models less trustworthy. That is a problem when we need to get the facts right. Large language models are not perfect. That is why hallucination is a concern for people who use large language models.

Large language models are really sensitive to what people tell them to do. Some people try to trick these models by putting instructions inside what they say. These bad instructions can tell the model to ignore the rules it was given before, or to share information, or to make something that can hurt people. The problem is that large language models read what people say one step at a time, and they do not have a way to know if what people say is good or bad.

This makes it hard to know if someone is really asking for help or if they are trying to trick the language model. Large language models have to deal with this problem because they are so sensitive to what people say, and they have to be careful with the instructions they get from people.

The main issue that this research is trying to solve is this:

3.1. How can we figure out when a language model is producing made-up information in time without changing how the model works on the inside?

3.2. How can we tell when someone is trying to trick the system with a prompt before they can actually do any harm?

3.3. How can we stop both of these problems from happening using one set of defences?[7]

To deal with these problems a new system is being suggested. This system has layers and can look at both the things people ask the language model and the answers it gives. The language model is the thing being talked about here and this system is trying to help the language model work better. The language model is what we are trying to improve. This new system is designed to work with the language model to make it more secure.

The new system is like a watchdog that checks on a language model. It does this without having to retrain the model. Instead, it adds some checks to make sure everything is okay before and after the model gives a response. [8]

The system has four parts:

Phase 1: Input Monitoring Layer

In this part, the system looks at what the user is asking before the language model gets it. It tries to find patterns that might be bad, like when someone's trying to trick the system or give it secret instructions. The system also looks for words that might be used to hurt it.. It does not just look for bad words. It also tries to understand what the user is really trying to do in case they are trying to be sneaky. This way, the system can stop people from doing things to it even if they do not use obvious bad words. [9]

Phase 2: Hallucination Detection Module

After the model generates a response, we check if the output makes sense. We use fact-checking methods like:
Checking important details against reliable sources

Making sure the response is logical

Analysing the responses' probability distribution to see if it is uncertain [10]

If we find inconsistencies or high uncertainty levels, we give the response a hallucination risk score.

The hallucination detection module helps ensure response accuracy by evaluating responses against trusted knowledge sources.

The hallucination detection module also evaluates responses for coherence.

The hallucination detection module measures response uncertainty.

Phase 3 is about the Risk Scoring Engine.
This is where we put together what we found from checking the input and detecting hallucinations.

We then calculate a Risk Scoring Engine score that takes into account a few things.

The Risk Scoring Engine uses things like how unusual the meaning's how wrong the facts are, and how likely it is that the instructions will be ignored.

All these things are considered when we calculate the Risk Scoring Engine score. [11]

Phase 4 is the Mitigation Layer.
The system has a Mitigation Layer for medium-risk cases and high-risk cases.

When the Mitigation Layer is activated, it uses mitigation strategies such as

Prompt sanitisation and reprocessing

Response regeneration, with constraints

Safe fallback responses when the verification of the Mitigation Layer fails. [12]

If we find inconsistencies or high uncertainty levels, we give the response a hallucination risk score. The hallucination detection module helps ensure response accuracy by evaluating responses against trusted knowledge sources. The hallucination detection module also evaluates responses for coherence. The hallucination detection module measures response uncertainty. [13]

This is where we put together what we found from checking the input and detecting hallucinations. We then calculate a Risk Scoring Engine score that takes into account things like how unusual the meaning is, how wrong the facts are and how likely it is that the instructions will be ignored. The Risk Scoring Engine uses these things when we calculate the Risk Scoring Engine score.

4. Result

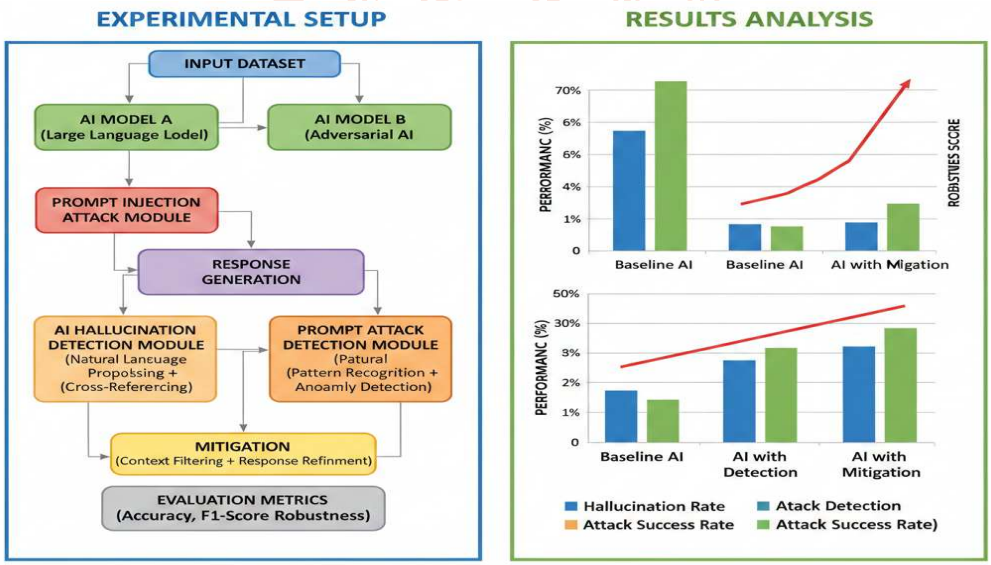

Figure 5: Experimental Setup & Result Analysis

5. Conclusion

Language models are being used in a lot of areas. We need to think about how reliable language models how secure they are. This research looked at two problems with modern AI systems. The first problem is when language models make things up. The second problem is when language models get tricked by inputs. When language models make things up, it can hurt how accurate they are. It can also hurt whether people trust language models. When language models get tricked, it can lead to bad things happening.[14]

To deal with these problems, this research suggested a way to detect and stop them that works outside of the language model. This new system checks what is going into the model, looks at what it means, checks if the model is making things up and has a way to stop things from happening. By checking both what is going in and what is coming out, this system helps keep the model reliable and secure.

We tested it with questions that have answers and with bad inputs that are designed to trick the model. The results showed that this system is better at detecting problems and does not make false alarms. It also showed that checking things in layers is better than checking one thing. This system does use a bit of computer power. It makes the model more robust and reliable.

In the future, we can make this system better by adding a way to check facts in time and by making it learn and adapt over time. We can also use this system to test how well it works against inputs and to see how it works in areas like healthcare and cybersecurity. Making AI systems trustworthy and secure is a challenge. We need to keep making the systems that protect them better. To deal with these problems, we need to switch from fixing things after they go wrong to stopping them from going wrong in the first place. [15]

The research showed that using layers of protection can really reduce the number of times bad inputs trick the model. The problem of AI systems making things up is still one, and we need to find a way to make them reason better and check facts outside of the model. We cannot just rely on the model itself to decide what is true. Looking ahead, we want to make sure that people can trust AI systems. As AI becomes more independent, we need to make sure that we have ways to check if they are working correctly and to stop them if they are not. The results of this research give us a plan to make AI systems more robust. The battle between people who make inputs and people who try to stop them is not over.

We need to keep working on making these protection systems so they can be used without slowing down the model. Large language models need to be reliable and secure. Large language models are used in various areas, and we need to think about how to make them more reliable and secure

6. References

[1] N. Varshney, W. Yao, H. Zhang, J. Chen and D. Yu, "A Stitch in Time Saves Nine: Detecting and Mitigating Hallucinations of Large Language Models," arXiv preprint arXiv: 2307.03987, 2023.

[2] P. Feldman, J. R. Foulds and S. Pan, "Trapping LLM Hallucinations Using Tagged Context Prompts," arXiv preprint arXiv:2306.06085, 2023.

[3] S. M. T. Tonmoy, S. M. Zaman, V. Jain et al., "A Comprehensive Survey of Hallucination Mitigation Techniques in Large Language Models," arXiv preprint arXiv:2301.05249, 2023.

[4] B. Peng, Z. Bi, Q. Niu et al., "Jailbreaking and Mitigation of Vulnerabilities in Large Language Models," arXiv preprint, 2024.

[5] S. Liu, Y. Gao, S. Li and P. Wang, "A Hallucination Detection and Mitigation Framework for Faithful Text Summarisation Using LLMs," Scientific Reports, vol. 16, 2026.

[6] D. Osmar and D. A. Dahl, "Prompt Injection Detection and Mitigation via AI Multi-Agent NLP Frameworks," arXiv preprint arXiv:2503.11517, 2025.

[7] J. Wei, X. Wang, D. Schuurmans et al., "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," Advances in Neural Information Processing Systems, 2022.

[8] P. Lewis, E. Perez, A. Piktus et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," Advances in Neural Information Processing Systems, 2020.

[9] T. Brown, B. Mann, N. Ryder et al., "Language Models are Few-Shot Learners," Advances in Neural Information Processing Systems, vol. 33, 2020.

[10] S. Bubeck, V. Chandrasekaran, R. Eldan et al., "Sparks of Artificial General Intelligence: Early Experiments with GPT-4," arXiv preprint arXiv:2303.12712, 2023.

[11] Y. Bai, A. Kadavath, S. Kundu et al., "Constitutional AI: Harmlessness from AI Feedback," arXiv preprint arXiv:2212.08073, 2022.

[12] E. Perez and I. Ribeiro, "Red Teaming Language Models to Reduce Harms: Methods, Scaling Behaviours and Lessons Learned," arXiv preprint arXiv:2209.07858, 2022.

[13] O. Ouyang, J. Wu, X. Jiang et al., "Training Language Models to Follow Instructions with Human Feedback," Advances in Neural Information Processing Systems, 2022.

[14] J. Deng, H. Zhang and Y. Liu, "Detecting Hallucinations in Large Language Models via Semantic Consistency Checking," Proceedings of the IEEE International Conference on Artificial Intelligence, 2024.

[15] Zou, Z. Wang, N. Carlini et al., "Universal and Transferable Adversarial Attacks on Aligned Language Models," arXiv preprint arXiv:2307.15043, 2023.