International Journal of Future Engineering Innovations

Training Time Vulnerabilities in Large Language Models: Data Poisoning and Backdoor Attacks

Ifeanyi Kingsley Egbuna [1*], Hanafi Musa Olayinka [2], Adegbola Bidemi Tijani [3], Saeed Hubairik Aliyu [4], Ann Ogechi Felix [5], OLUWSOLA Abiodun Elijah [6], Mathew Ayokunle Alabi [7]

[1] Department of Supply Chain Management, Marketing, and Management, Wright State University, Dayton, USA
[2] Department of Computer science & Engineering Technology, University of Houston Downtown, United States
[3] Department of Electrical and Electronic Engineering, University of Ibadan, Nigeria
[4] Department of Mechatronics and Robotics Engineering, South Ural State University, Russia
[5] School of Architecture, Computing and Engineering, University of East London, United Kingdom
[6] Department of Mechanical Engineering, University of Ilorin, Nigeria
[7] Department of Computer Science and Engineering, Obafemi Awolowo University, Nigeria

* Corresponding Author: **Ifeanyi Kingsley Egbuna**

Article Info

ISSN (online): 3049-1215
Volume: 02
Issue: 03
May-June 2025
Received: 06-03-2025
Accepted: 08-04-2025
Page No: 60-68

Abstract
Large Language Models (LLMs) have revolutionized natural language processing tasks across diverse domains. However, their increasing complexity and reliance on massive training datasets have introduced novel security risks. This paper explores training time vulnerabilities in LLMs, with a focus on data poisoning and backdoor attacks. Data poisoning involves injecting malicious samples into the training data to subtly influence model behavior, while backdoor attacks embed hidden triggers that cause the model to behave maliciously only under specific conditions. We analyze the mechanics, potential impacts, and detection challenges associated with these threats, highlighting their implications for model integrity, trustworthiness, and deployment safety. Finally, we review recent defense strategies and propose future research directions to mitigate these emerging risks.

DOI: https://doi.org/10.54660/IJFEI.2025.2.3.60-68

Keywords: Trustworthy AI, Large Language Models (LLMs), Data Poisoning, Backdoor Attacks, Training Time Vulnerabilities,

1. Introduction

Large Language Models (LLMs) have become a cornerstone in the evolution of artificial intelligence (AI), significantly advancing the field of natural language processing (NLP). (Algahtani *et al*., 2023). These models, powered by deep learning techniques and vast amounts of data, have been at the forefront of applications ranging from machine translation and content generation to sentiment analysis and question answering. (Razzaq & Shah 2025) [13]. LLMs such as OpenAI's GPT series and Google's BERT have demonstrated remarkable capabilities in understanding and generating human like text, revolutionizing industries such as healthcare, finance, and customer service. As the scale of these models continues to grow exemplified by GPT 3's 175 billion parameters so too does their potential to transform various aspects of human computer interaction (Brown *et al*., 2020) [5]. The complexity and sheer size of LLMs have, however, introduced novel challenges, particularly in the domain of security. Despite their success, the increasing reliance on large scale datasets for training these models has revealed significant vulnerabilities. In particular, the process of training LLMs often involving massive amounts of uncurated data has made them susceptible to subtle forms of manipulation that can have far reaching consequences. (Al-karusi *et al*., 2024).

Among these vulnerabilities, data poisoning and backdoor attacks have emerged as two of the most concerning threats to the integrity of these models. Data poisoning occurs when malicious actors deliberately insert harmful data into the training set to influence the model's behavior in a way that aligns with their objectives (Biggio *et al*., 2012) [3]. This form of attack can alter the model's outputs in subtle yet harmful ways, leading to performance degradation or skewed results. On the other hand, backdoor attacks involve embedding hidden triggers within the training data that cause the model to behave maliciously only when certain conditions are met (Gu *et al*., 2017) [10]. For example, a backdoor might make a model classify images or text incorrectly when a specific pattern or "trigger" is input, posing severe risks when deployed in sensitive applications.

The problem of training time vulnerabilities in LLMs is particularly pressing because, unlike attacks that occur post deployment, these vulnerabilities are introduced during the training phase. This makes detection and mitigation significantly more difficult. Moreover, the high degree of automation and the lack of transparency in model training processes further exacerbate the challenges in identifying these vulnerabilities.

The integration of large, unverified datasets often scraped from the internet adds another layer of complexity, as it becomes nearly impossible to manually review all data inputs. Thus, while LLMs are celebrated for their ability to generate coherent and contextually relevant text, the risks associated with their training processes are often overlooked. Addressing these vulnerabilities is crucial for maintaining the integrity and safety of AI systems. If left unaddressed, data poisoning and backdoor attacks can undermine the trustworthiness of LLMs, leading to unintended consequences in real world applications. For instance, a backdoor attack might allow a malicious actor to trigger harmful actions in a deployed system, causing the model to behave unpredictably or perform illicit actions. Similarly, data poisoning can subtly degrade model performance, leading to biased or inaccurate outputs that could have serious repercussions, especially in high stakes domains such as medical diagnosis or financial forecasting. Furthermore, the lack of robust security measures could erode public trust in AI technologies, hindering their widespread adoption and limiting their potential benefits. Thus, ensuring that LLMs are secure during their training phase is essential not only for the performance and reliability of the models themselves but also for the broader societal trust in AI systems.

The objective of this paper is to explore the training time vulnerabilities of LLMs, specifically focusing on data poisoning and backdoor attacks. By examining the mechanisms behind these threats, we aim to shed light on their potential impact on model performance and security. This paper also seeks to address the challenges involved in detecting these vulnerabilities, considering the complexity of LLM training processes and the difficulties inherent in identifying malicious data patterns. Additionally, we review existing defense strategies and propose future research directions aimed at mitigating these emerging risks. By contributing to the understanding of training time vulnerabilities in LLMs, this paper seeks to enhance the robustness of AI systems and promote the development of more secure, trustworthy models.

2. Literature Review

Large Language Models (LLMs) are a class of deep learning models designed to understand, generate, and manipulate human language. These models, particularly those based on transformer architectures such as GPT 3 (Brown *et al*., 2020) [5] and BERT (Devlin *et al*., 2018), have reshaped natural language processing (NLP) by achieving state of the art results across a wide range of tasks. LLMs are typically trained on vast, unstructured datasets scraped from the web, which include books, articles, websites, and other textual sources. These models are built on neural networks, specifically using a transformer architecture, which relies on self attention mechanisms to process large amounts of text data in parallel. The training process for LLMs involves exposing the model to billions or even trillions of words, allowing it to learn the statistical relationships between different words, sentences, and concepts.

The success of LLMs lies in their ability to capture contextual relationships in text, enabling them to perform tasks such as text generation, sentiment analysis, machine translation, question answering, and even dialogue systems (Vaswani *et al*., 2017) [16]. Their applications are diverse and have penetrated numerous industries, including healthcare, where they assist in medical diagnostics and patient care; in customer service, where they power chatbots and virtual assistants; and in the entertainment industry, where they aid in content creation and interactive storytelling. As LLMs continue to improve in both scale and performance, their societal impact grows, positioning them as transformative tools for both businesses and consumers. However, the scale at which these models operate introduces challenges, particularly in the realm of security. The need for large and diverse datasets to train LLMs often leads to the use of data scraped from the internet, which, while expansive, can introduce noise and biases into the models. As LLMs evolve, their susceptibility to subtle manipulations in the training data also grows, raising concerns about their trustworthiness and robustness.

Data poisoning and backdoor attacks represent two of the most pressing threats to the integrity of machine learning models, including LLMs. Data poisoning occurs when an adversary deliberately injects malicious data into the training set to degrade the model's performance or alter its behavior in a specific direction. This type of attack can be difficult to detect, as the poisoned data may closely resemble legitimate training data but contain small, targeted perturbations that lead the model to misclassify certain inputs or behave unpredictably under certain conditions. Early work by Biggio *et al*. (2012) [3] introduced the concept of poisoning attacks and demonstrated how attackers could manipulate the training data of classifiers to cause misclassification. In the context of LLMs, data poisoning could manifest in various forms, such as inserting biased or misleading information, which could skew the model's outputs or cause it to produce harmful or inaccurate results (Munoz Gonzalez *et al*., 2017) [12].

In contrast, backdoor attacks involve embedding hidden triggers or patterns within the training data that cause the model to behave inappropriately only when these triggers are encountered during inference. For example, an attacker might insert specific keywords or phrases into the training set that, when later encountered by the model, cause it to generate

biased, harmful, or incorrect outputs (Gu *et al*., 2017) [10]. These attacks are particularly insidious because the model behaves normally under most conditions but exhibits malicious behavior when the backdoor trigger is activated. Recent studies on backdoor attacks, such as those by Liu *et al*. (2018) [11], highlight the vulnerability of deep learning models, including LLMs, to these types of adversarial manipulations. These attacks can be particularly devastating when deployed in critical applications such as autonomous systems or healthcare diagnostics, where small, undetected changes in behavior can lead to catastrophic outcomes.

Both data poisoning and backdoor attacks have been shown to significantly undermine the performance and reliability of machine learning models, especially when the attacker has sufficient control over the training data. These types of attacks are particularly challenging to defend against because they exploit the inherent vulnerability of models to imperfect or untrusted training data. Moreover, the sheer scale and complexity of LLMs, which are typically trained on datasets consisting of billions or even trillions of words, make it exceedingly difficult to identify and eliminate malicious influences from the data without comprehensive monitoring and validation.

The scale and complexity of LLMs inherently make them more vulnerable to attacks during the training phase. Traditional machine learning models, especially those with smaller datasets and simpler architectures, are relatively easier to monitor and audit for potential security threats. In contrast, LLMs are trained on vast datasets sourced from the internet, a domain where data quality and integrity are difficult to guarantee. The data scraped from the web is often unfiltered and noisy, containing biases, inaccuracies, and potentially harmful content. Furthermore, the training process of LLMs is computationally expensive and time consuming, often involving thousands of GPUs over weeks or even months. This massive scale presents unique challenges in ensuring the integrity of the training data and the resulting model outputs.

The introduction of adversarial data, in the form of either poisoned data or backdoor triggers, can subtly alter the behavior of LLMs without immediate detection. For instance, an attacker might craft a small set of inputs with hidden triggers that, when embedded in a massive dataset, would be virtually impossible to spot by traditional methods. The size of LLMs, along with the diverse nature of their training data, complicates efforts to identify such vulnerabilities early in the training process. Additionally, because LLMs are trained to generalize across a wide variety of tasks, the presence of malicious data in one part of the training set could result in unintended consequences across a broad range of applications.

Recent research has pointed out that as LLMs become more complex, they are also more susceptible to "distributional shifts" in their training data, which can be exploited through data poisoning and backdoor attacks (Carlini *et al*., 2020). With LLMs being increasingly used in real world applications, such as healthcare, autonomous vehicles, and financial systems, these attacks can have severe consequences, including loss of privacy, financial damage, and compromised decision making. Furthermore, the complexity of LLMs makes it difficult to interpret their decision making process, often leading to what is known as the "black box" problem, where even the model developers cannot fully explain how certain predictions or outputs are made. This lack of transparency complicates efforts to audit the models for potential vulnerabilities, making LLMs even more prone to adversarial exploitation.

As these models continue to grow, both in terms of their size and their deployment in critical areas, the importance of addressing training time vulnerabilities becomes ever more apparent. Detecting and mitigating data poisoning and backdoor attacks in LLMs will require new methods, including advanced techniques in data verification, model transparency, and adversarial testing. Without these measures, LLMs will continue to pose significant security risks, undermining their potential to be used safely in high stakes environments.

The field of LLMs has made tremendous strides in advancing NLP applications, but these models' increasing complexity and scale have opened up new avenues for adversarial manipulation during training. As explored in the existing literature, both data poisoning and backdoor attacks present significant challenges to the integrity and security of LLMs. Given the widespread use and impact of LLMs in society, addressing these vulnerabilities is essential for ensuring the safe and ethical deployment of these models. Future research must continue to focus on developing robust defense mechanisms and detection techniques to safeguard the next generation of language models from adversarial threats during the critical training phase.

3. Discussion

Data poisoning in large language models

Data poisoning refers to the deliberate manipulation of a machine learning model’s training data to cause the model to behave in an adversarial manner. In the context of Large Language Models (LLMs), data poisoning typically involves inserting malicious, misleading, or incorrect samples into the vast datasets used to train these models. The poisoned data is crafted to subtly influence the model’s behavior without immediately compromising its general performance. Instead of causing catastrophic failure, data poisoning typically aims to induce biases, misclassifications, or certain undesirable behaviors under specific conditions.

In LLMs, the mechanism of data poisoning is most effective when the malicious data is injected into large training sets, making it hard to detect or isolate. The sheer volume of training data often consisting of billions or trillions of words makes manual inspection impractical. Poisoned data can be designed in several ways, from introducing bias in text generation to subtly altering the model’s understanding of certain terms or concepts. For instance, an attacker could insert phrases or sentences that distort the model’s ability to understand context, thereby producing biased or discriminatory outputs when the model encounters similar contexts during deployment (Munoz Gonzalez *et al*., 2017) [12].

The primary goal of data poisoning is to exploit the model’s sensitivity to the patterns present in the training data. By embedding specific patterns or manipulating statistical relationships in the data, attackers can influence the model's behavior without altering its overall structure. This subtlety is what makes data poisoning particularly dangerous, as the model may continue to perform well in general tasks but fail when exposed to specific situations where the poisoned data has an effect.

Types of poisoning attacks

Data poisoning attacks can take many forms, each with its own unique approach and impact on the model's training process. The most common types of poisoning attacks include label flipping, targeted poisoning, and gradient based poisoning.

Label Flipping: In this attack, the labels of specific training samples are altered. For example, in a sentiment analysis task, positive reviews could be relabeled as negative, or vice versa. This type of attack aims to confuse the model during training by altering the ground truth of certain data points. For LLMs, this could mean subtly altering the meaning of words or phrases, leading the model to learn incorrect associations. Label flipping can have a significant impact on a model's performance, especially if it occurs frequently in the training data (Biggio *et al*., 2012) [3].

Targeted Poisoning: Targeted poisoning is a more sophisticated attack that manipulates the training data to influence the model's performance on a particular input or task. For example, an attacker might insert a series of training samples designed to make the model misinterpret certain keywords or phrases. The goal of targeted poisoning is not to degrade the overall performance of the model but to subtly skew its output in specific ways. In LLMs, this could involve inserting sentences that manipulate the model's response to particular queries, leading to biased or inaccurate outputs when the model is deployed (Munoz Gonzalez *et al*., 2017) [12].

Backdoor Poisoning: While closely related to backdoor attacks, backdoor poisoning involves inserting specific "trigger" phrases or patterns in the training data that cause the model to misbehave only under specific conditions. For instance, the attacker could insert benign looking text with a hidden trigger such as a certain phrase or keyword that causes the model to produce biased or harmful outputs when the trigger is encountered during inference. Unlike traditional backdoor attacks, backdoor poisoning focuses on subtly embedding these triggers in the training data, making it harder to detect during the training phase (Gu *et al*., 2017) [10].

Data duplication and perturbation: In this attack, malicious actors may duplicate benign examples in the dataset or add small perturbations to existing samples. These changes might seem inconspicuous, but they can be enough to distort the model's understanding of certain patterns. For LLMs, this could include subtle shifts in sentence structure or the introduction of uncommon phrases that could, over time, cause the model to misinterpret or misgenerate certain types of text (Carlini *et al*., 2020).

3.1 Impact of Poisoning

The consequences of data poisoning on LLMs can be profound, affecting various aspects of the model's performance, reliability, and fairness. Even though the model might perform adequately on a majority of tasks, poisoned data can introduce serious flaws in specific domains or under particular conditions.

Accuracy Degradation: One of the most immediate impacts of data poisoning is a decrease in the model's overall accuracy. Since LLMs are trained to generalize from the data they are provided, even a small amount of poisoned data can result in misclassifications or incorrect predictions, especially when the poisoned data reflects a consistent bias. For instance, an LLM trained on poisoned data that misrepresents certain topics or phrases might generate inaccurate text or provide wrong answers to specific questions. This can lead to user frustration and a loss of confidence in the model.

Reliability and robustness issues: Poisoned data can also reduce a model's reliability by making it more sensitive to specific inputs. LLMs, by design, aim to generalize from their training data, but poisoned data can cause the model to fail under certain conditions. For example, an LLM trained with biased or maliciously altered data might exhibit erratic or unreliable behavior when asked about controversial or sensitive topics. This becomes particularly problematic in applications where reliability is critical, such as in healthcare or legal systems, where a small misstep could have serious consequences.

Fairness Concerns: Data poisoning can exacerbate fairness issues in LLMs. Since these models often learn to mimic the biases and prejudices present in their training data, poisoning attacks that manipulate this data could lead to outputs that are discriminatory, biased, or otherwise unethical. For example, if poisoned data injects biased language or unfair stereotypes, the model might perpetuate these biases when generating responses, leading to unequal treatment of certain groups. Such issues are especially problematic when LLMs are deployed in settings that require fairness and equity, such as hiring systems, customer service, or healthcare diagnostics (Binns *et al*., 2018) [4].

Model Trustworthiness: Perhaps one of the most insidious impacts of data poisoning is its potential to undermine the trustworthiness of the model. Users are less likely to trust a model that produces unexpected or biased results, especially when these issues arise from training data that is difficult to audit. The lack of transparency in LLMs, often referred to as the "black box" nature of these models, makes it particularly challenging to understand how or why certain decisions were made, further complicating efforts to identify and address poisoning attacks.

3.2 Challenges in Detection

Detecting data poisoning in LLMs is a particularly difficult task, as it often involves identifying small, subtle changes in the training data that are designed to evade detection. The large and diverse datasets used to train LLMs make it difficult to manually inspect all training samples for anomalies or malicious content. Additionally, traditional methods of data validation, such as manual curation or expert review, are impractical when dealing with datasets that consist of billions of words.

Scale of Data: The scale of LLM training datasets is one of the biggest challenges in detecting poisoned data. For example, models like GPT 3 are trained on datasets that span hundreds of billions of words, making it nearly impossible to spot individual instances of poisoned data by hand. Traditional data validation methods, which work well for smaller datasets, become ineffective in the context of such

large scale training.

Subtlety of the poisoning: Unlike more overt forms of attack, where adversaries may introduce easily detectable anomalies, data poisoning often involves subtle changes to the training data. The poisoned data may appear to be legitimate, and its effects may not be immediately apparent until the model is deployed in real world scenarios. This subtlety complicates efforts to detect and isolate the poisoned data.

Lack of ground truth: Another challenge in detecting poisoned data is the lack of a definitive "ground truth" for many LLM tasks. In many NLP applications, such as text generation, sentiment analysis, or summarization, there is no clear, universally accepted "correct" output. This ambiguity makes it difficult to determine whether a model's behavior is the result of malicious manipulation or simply an inherent limitation of the model's generalization capabilities.

Overfitting and generalization: Finally, because LLMs are trained to generalize across a wide range of tasks, they may overfit to the poisoned data without showing signs of direct performance degradation. Overfitting occurs when the model learns specific patterns from the poisoned data that don't generalize well to real world inputs, resulting in subtle misclassifications or behavior anomalies that are hard to trace back to the training data.

Data poisoning presents a significant challenge to the security and reliability of Large Language Models. By subtly manipulating the training data, adversaries can degrade the accuracy, fairness, and reliability of LLMs in ways that are difficult to detect. As LLMs continue to grow in size and complexity, the risks associated with data poisoning become more pronounced. Addressing these risks requires the development of robust methods for detecting and mitigating poisoned data, as well as strategies for ensuring the integrity of training datasets in the first place. Without such measures, LLMs will remain vulnerable to adversarial attacks, with potentially far reaching consequences for their deployment in critical real world applications.

3.3 Backdoor attacks in large language models

A backdoor attack refers to the deliberate insertion of hidden triggers or patterns into the training data of a machine learning model, which then causes the model to behave maliciously or unpredictably when these triggers are encountered during inference. In the context of Large Language Models (LLMs), backdoor attacks typically involve embedding specific sequences of words, phrases, or other subtle patterns in the training data that, when activated, force the model to generate biased, harmful, or incorrect outputs. The core mechanism behind backdoor attacks lies in the ability of the attacker to manipulate the model's behavior in a targeted manner, without altering its performance on the majority of tasks. This allows the model to appear functional and reliable for most use cases, but once exposed to a particular trigger, it exhibits malicious behavior. Backdoor attacks are particularly dangerous because they often remain undetected until the model is deployed in real world scenarios, where the attacker can activate the hidden trigger.

For LLMs, backdoor attacks can be particularly effective due to the size and complexity of their training datasets. These models are typically trained on vast amounts of text data scraped from diverse sources across the web. With such large and varied datasets, subtle manipulations can easily go unnoticed. An attacker may introduce a small set of inputs that contain an embedded backdoor trigger, and when this trigger is encountered during testing or deployment, the model will produce outputs that reflect the attacker's intent, such as bias, misinformation, or harmful content. The challenge in detecting these attacks lies in the difficulty of identifying the specific trigger and the intricacy of the model's decision making process, which is often opaque due to the "black box" nature of deep learning systems.

Backdoor attacks can manifest in numerous scenarios, and their subtle nature makes them difficult to detect and mitigate. One example is the manipulation of text generation in an LLM designed for conversational agents. Suppose an attacker inserts specific trigger phrases like "I think it's time for a change" into the training data, with the intent that these phrases will cause the model to generate harmful or inappropriate responses when encountered. During normal operation, the model would produce coherent and safe responses, but when the phrase "I think it's time for a change" appears in a conversation, it might trigger the model to provide malicious advice, spread misinformation, or promote harmful behavior.

Another example is in sentiment analysis or opinion based tasks. An attacker could inject poisoned data into the training set, containing phrases that cause the model to misclassify particular opinions or statements as positive or negative when certain keywords are used. For instance, a review of a controversial product could be altered in such a way that when specific words appear such as "test" or "study" the model rates the review as overwhelmingly positive, even though the content of the review is critical. This manipulation could skew the results of sentiment analysis in business applications, making it difficult to accurately gauge consumer sentiment.

In more targeted attacks, backdoors can be inserted into systems that rely on LLMs for decision making. For instance, an autonomous vehicle using an LLM for natural language understanding could be compromised by embedding backdoor triggers in the model's training data. These triggers could cause the vehicle to misinterpret certain commands or instructions, potentially leading to unsafe driving behaviors when the model encounters specific phrases or situations. Such attacks could be disastrous in high stakes environments like autonomous vehicles, healthcare systems, or financial trading systems, where small errors can have severe consequences.

3.4 Impact of backdoor attacks

The impact of backdoor attacks in LLMs can be wide ranging, posing significant risks to the safety, fairness, and reliability of AI systems. When a backdoor trigger is activated, the model may produce outputs that are biased, discriminatory, or harmful, undermining the integrity of the system. In applications like sentiment analysis or content moderation, this could result in the amplification of harmful stereotypes or the spreading of false or misleading information. In customer service applications, a backdoor attack could cause a chatbot to behave unethically, providing inappropriate responses to users.

The potential harms of backdoor attacks are particularly concerning in fields where decision making needs to be transparent, accountable, and fair. In sectors like healthcare,

where LLMs are used to provide medical advice or assist with diagnostic tools, a backdoor could lead to catastrophic errors, such as recommending unsafe treatments or misdiagnosing patients. In financial sectors, LLMs could be compromised to give biased investment recommendations, influencing financial decisions in harmful ways. Moreover, backdoor attacks can undermine trust in AI systems. When users realize that an LLM can produce malicious outputs under certain conditions, they may lose confidence in the technology, reducing its effectiveness and adoption in critical applications.

Another potential harm is the manipulation of public opinion, especially when LLMs are used to generate social media content, news articles, or other forms of public communication. If an attacker can activate a backdoor that skews the content in a specific direction, this could contribute to the spread of misinformation, political manipulation, or biased narratives. In such cases, the backdoor attacks could have far reaching societal consequences, making it essential to develop robust detection and defense strategies.

3.5 Challenges in Detection

Detecting backdoor attacks in LLMs is especially difficult due to several factors inherent in their design and deployment. First, LLMs operate as "black box" models, meaning that their internal decision making processes are not easily interpretable by humans. This lack of transparency makes it challenging to understand why a model behaves in a certain way, especially when it produces an unexpected output triggered by a backdoor. For example, if a model generates biased or harmful content when specific phrases are input, it can be difficult to trace whether this behavior is due to a backdoor or simply a result of the model's training data or structure.

Second, the subtlety of backdoor attacks further complicates detection. Unlike traditional adversarial attacks, which may introduce more overt anomalies in the model's performance, backdoor triggers are typically designed to remain dormant until specific conditions are met. As a result, the model's performance may remain largely unaffected during routine testing, making it difficult to detect the presence of a backdoor through conventional evaluation methods. Even if testing is conducted using a range of inputs, the backdoor trigger may not be activated, leaving the model's behavior seemingly unaltered.

Additionally, the sheer scale of data used to train LLMs increases the complexity of detecting backdoor triggers. Large datasets often contain a high level of noise and variation, making it hard to identify malicious alterations. Furthermore, backdoor triggers are typically embedded in small portions of the data, making them hard to spot in the context of billions of words. Even sophisticated machine learning based detection techniques may struggle to differentiate between normal patterns in the data and maliciously injected triggers.

Finally, the deployment of LLMs in real world applications often involves continuous learning or fine tuning based on user interactions or additional data. This presents a unique challenge, as backdoor triggers could be introduced or activated during the model's operational phase, making it difficult to catch them in pre deployment testing. This dynamic nature of LLMs means that backdoor attacks can evolve over time, further complicating efforts to ensure long term security.

Backdoor attacks represent a significant security threat to the integrity and safety of Large Language Models. These attacks exploit subtle manipulations of the training data to induce harmful or biased behavior when specific triggers are encountered. While LLMs generally exhibit high performance, their susceptibility to backdoor attacks introduces risks that can affect a wide range of applications, from autonomous systems to content moderation and healthcare. The challenges in detecting backdoor attacks are compounded by the black box nature of LLMs, the subtlety of the attacks, and the scale of the data they are trained on. To mitigate the risks posed by backdoor attacks, researchers and practitioners must continue to develop advanced detection methods, as well as defense mechanisms that ensure the trustworthiness of LLMs in real world applications.

3.6 Detection and mitigation strategies

Techniques for detecting data poisoning and backdoor attacks

Detecting data poisoning and backdoor attacks in Large Language Models (LLMs) is a critical challenge due to the subtlety and complexity of these attacks. However, several techniques have been proposed to identify training time vulnerabilities, each focusing on different aspects of model behavior and data integrity.

Anomaly Detection: One common approach to detecting data poisoning is the use of anomaly detection techniques. These methods aim to identify outliers or irregular patterns in the data that could indicate the presence of maliciously inserted samples. For example, unsupervised anomaly detection algorithms can be used to compare training data against a baseline, flagging samples that deviate from typical patterns (Ruff *et al*., 2018) [14]. While effective in smaller scale datasets, anomaly detection methods can be less effective in large scale LLM training due to the sheer volume of data involved.

Behavior Analysis: Another promising approach is analyzing the behavior of models during training and after deployment. Researchers have proposed behavior based detection methods where the model's predictions are monitored for inconsistencies or abnormal outputs when exposed to certain inputs. For instance, if a model begins to produce biased or harmful content in response to seemingly benign triggers, this could indicate the presence of a backdoor. Behavior analysis can also involve observing shifts in model performance or identifying cases where the model performs unusually well or poorly with particular inputs. This technique is particularly useful for identifying backdoor attacks since the malicious behavior often becomes apparent only when specific triggers are encountered (Zhu *et al*., 2019) [18].

Testing Under Varied Conditions: A more comprehensive detection strategy involves testing the model under a variety of conditions to observe how it behaves with different types of inputs. This includes testing edge cases, adversarial examples, or inputs that resemble potential triggers of backdoor attacks. The idea is to identify scenarios where the model's behavior deviates from expected outcomes, suggesting that hidden backdoor triggers might be present. Such testing, however, is computationally expensive and may not always be feasible for large scale LLMs, where the input

space is vast, and triggers are often subtle and varied.

3.7 Defensive Mechanisms

Several strategies have been proposed to defend against data poisoning and backdoor attacks, aiming to either make the training process more robust or to detect and remove malicious data before it affects the model.

Robust training methods: One defense strategy against data poisoning involves robust training techniques, which aim to make the model less sensitive to poisoned data. These techniques include methods like *outlier detection during training* or *data augmentation*, which help the model learn more generalized features that are less susceptible to poisoning. For example, adversarial training, a technique commonly used for improving model robustness to adversarial attacks, can also be applied to defend against data poisoning. In this method, the model is exposed to adversarial examples during training, encouraging it to learn more robust features that are resistant to malicious manipulation (Shafahi *et al*., 2018).

Anomaly detection in the dataset: Another defensive mechanism involves pre processing the training data to identify and remove poisoned samples before they are used to train the model. This can involve using statistical or machine learning techniques to identify anomalies or inconsistencies in the dataset that could indicate poisoning. Methods such as clustering, nearest neighbor analysis, and outlier detection algorithms can be applied to detect data that deviates from normal patterns. In addition, some methods propose a multi step verification process, where data is first labeled or categorized before training, and potential outliers are flagged for further inspection (Buurman *et al*., 2019) [6].

Data sanitization techniques: Data sanitization refers to techniques designed to cleanse the training data of malicious or misleading samples. This can involve removing samples that have been identified as outliers or implementing more sophisticated filtering techniques to eliminate content that could potentially contain hidden triggers for backdoor attacks. Data sanitization can be combined with other methods, such as anomaly detection or robust training, to strengthen defenses against data poisoning and backdoor attacks. However, data sanitization is not foolproof, as sophisticated attackers may find ways to evade detection or insert poisoned data that appears benign (Zhang *et al*., 2020).

3.8 Limitations of current approaches

While the current techniques for detecting and mitigating data poisoning and backdoor attacks have made significant strides, they are not without limitations, particularly when applied to large scale LLMs.

Scalability Issues: One of the biggest challenges with existing detection methods is their scalability. Many techniques, such as anomaly detection and testing under varied conditions, are computationally intensive and may not be feasible when applied to the enormous datasets required for training LLMs. With models trained on hundreds of billions of words, running exhaustive tests or detecting anomalies in every sample becomes impractical. As a result, current methods are often only effective on smaller models or datasets.

Subtle nature of attacks: The subtlety of backdoor and data poisoning attacks makes them hard to detect, especially in the absence of clear, obvious signs of manipulation. In many cases, poisoned data or backdoor triggers do not immediately degrade the model's overall performance. This means that conventional detection methods, which focus on overall accuracy or performance metrics, may fail to identify vulnerabilities that only emerge under specific conditions. Additionally, attackers can design backdoor triggers that blend seamlessly with the training data, making it difficult for even advanced detection methods to spot them.

Evolving Attacks: As models continue to evolve, so too do the methods employed by attackers. New backdoor and poisoning strategies are constantly being developed, and it can be difficult for current defense mechanisms to keep up with these evolving threats. For instance, attackers may use more sophisticated techniques to hide backdoor triggers or poison data in ways that evade traditional detection methods. This dynamic nature of attacks means that researchers and practitioners must continually update their defenses to stay ahead of potential threats.

4. Future research directions

4.1 Improving detection methods

One of the key areas for future research is the development of more effective detection methods. Traditional anomaly detection and testing under varied conditions will need to be augmented with more sophisticated techniques, such as *unsupervised learning* for anomaly detection, or *adversarial training* techniques, which can be used to identify poisoned data before it corrupts the model (Goodfellow *et al*., 2014). Adversarial training involves exposing the model to adversarial examples during training, allowing it to learn to identify and resist adversarial inputs, including poisoned data. Furthermore, hybrid models that combine multiple detection techniques such as statistical analysis, behavior-based analysis, and deep learning-based anomaly detection could offer more robust defense mechanisms.

Another promising direction is the integration of *unsupervised learning* to better identify anomalous data. Unsupervised methods do not rely on labeled data and can potentially identify poisoned data that might not fit into predefined categories. This approach could be particularly valuable for large scale LLMs, where it may be difficult to label all data manually.

4.2 Enhancing model robustness

Beyond detection, another critical area for research is the development of LLM architectures that are more robust to data poisoning and backdoor attacks. For example, model architectures could be designed to better detect and mitigate the influence of poisoned data during training. This might involve improving the model's ability to generalize by increasing its capacity to detect outliers and unusual patterns, which would help it resist malicious data manipulations. Additionally, integrating *defensive regularization* techniques, such as *mixup* or *label smoothing*, during training could help improve the robustness of LLMs against both data poisoning and backdoor attacks (Zhang *et al*., 2017) [10].

Another area for improving model robustness is developing models that are less reliant on the training data itself. By focusing on the model's intrinsic properties, such as robustness to adversarial inputs or stronger generalization

capabilities, researchers may be able to reduce the impact of malicious data.

Addressing training time vulnerabilities in AI models requires more than just technical solutions; it also necessitates strong policy frameworks and ethical considerations. As AI models become increasingly integrated into society, the potential for misuse grows, and it is essential to establish ethical guidelines and regulations to prevent and mitigate the consequences of malicious attacks. Research into the ethical implications of backdoor and poisoning attacks should explore how to ensure accountability in AI systems and develop safeguards that protect vulnerable sectors, such as healthcare, finance, and law enforcement, from the consequences of compromised models.

Furthermore, policymakers will need to establish regulations that require transparency in AI model development, particularly in the context of LLMs. This could involve mandates for routine audits of training data, validation procedures, and post deployment monitoring to ensure that AI models remain secure and trustworthy throughout their lifecycle.

As Large Language Models continue to play an integral role in various applications, it is imperative to address the training time vulnerabilities that arise from data poisoning and backdoor attacks. While current detection and mitigation techniques have made progress, more work is needed to create scalable, effective defense mechanisms for large scale models. Future research will need to focus on improving detection methods, enhancing model robustness, and establishing comprehensive policy frameworks to ensure that these AI systems remain secure, fair, and reliable in real world applications.

5. Conclusion

As Large Language Models (LLMs) continue to evolve and become integral to various applications across industries, addressing the security vulnerabilities that arise during the training phase is becoming increasingly critical. Data poisoning and backdoor attacks pose significant risks to the integrity and safety of AI systems, particularly in high stakes environments such as healthcare, finance, and autonomous systems. These attacks, often subtle and sophisticated, can manipulate a model's behavior in ways that are difficult to detect, undermining user trust and potentially causing severe harm.

The techniques for detecting and mitigating data poisoning and backdoor attacks have made notable advancements, including anomaly detection, behavior analysis, and testing under varied conditions. However, these methods still face challenges in scalability, precision, and adaptability to evolving attack strategies. Existing defense mechanisms, such as robust training methods, anomaly detection in datasets, and data sanitization, provide a solid foundation, but they remain insufficient for large scale LLMs, which are particularly vulnerable due to their complexity and the vast amounts of data used for training.

Future research directions must focus on improving detection techniques, particularly by incorporating adversarial training, unsupervised learning, and more sophisticated anomaly detection algorithms. Additionally, enhancing model robustness by developing architectures less susceptible to data poisoning and backdoor attacks is a crucial avenue for advancing the security of LLMs. Beyond technical solutions, policy frameworks and ethical guidelines will be essential to ensure that AI models are developed, deployed, and monitored in a way that prioritizes security, transparency, and fairness.

The growing complexity and capabilities of LLMs necessitate an ongoing effort to develop more effective methods for securing these models. Addressing training time vulnerabilities is not only essential for safeguarding the performance and safety of AI systems but also for ensuring their broader societal acceptance and trust. The collaboration between researchers, practitioners, policymakers, and ethicists will be crucial in building AI systems that are both powerful and secure, capable of advancing the potential of artificial intelligence while minimizing the risks posed by malicious manipulation.

6. References

1. Al-Kharusi Y, Khan A, Rizwan M, Bait-Suwailam MM. Open-Source Artificial Intelligence Privacy and Security: A Review. Computers. 2024;13(12):311. https://doi.org/10.3390/computers13120311
2. Alqahtani T, Badreldin HA, Alrashed M, Alshaya AI, Alghamdi SS, Bin Saleh K, Aljaber A, Amer YS, Harbi HA, Albekairy AM. The emergent role of artificial intelligence, natural learning processing, and large language models in higher education and research. Research in Social and Administrative Pharmacy. 2023;19(8):1236-1242. https://doi.org/10.1016/j.sapharm.2023.06.012
3. Biggio B, Fumera G, Roli F. Poisoning Attacks in Data Mining. ACM Computing Surveys. 2012;44(3):1-36. https://doi.org/10.1145/2187671.2187676
4. Binns R, Garfinkel S, Mohta A. Discrimination in online ad delivery. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems; 2018 Apr 21-26; Montreal, Canada. New York: ACM; 2018. p. 1-13. https://doi.org/10.1145/3173574.3173951
5. Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, Agarwal S, Herbert-Voss A, Krueger G, Henighan T, Child R, Ramesh A, Ziegler DM, Wu J, Winter C, Hesse C, Chen M, Sigler E, Litwin M, Gray S, Chess B, Clark J, Berner C, McCandlish S, Radford A, Sutskever I, Amodei D. Language Models are Few-Shot Learners. In: Advances in Neural Information Processing Systems 33 (NeurIPS 2020); 2020 Dec 6-12; Virtual Conference. Red Hook, NY: Curran Associates; 2020. p. 1877-1901.
6. Buurman B, Sculley D, Grittner J. Detecting data poisoning in machine learning systems. In: Proceedings of the IEEE/ACM International Conference on Automated Software Engineering; 2019 Nov 11-15; San Diego, USA. New York: IEEE; 2019. p. 1-10. https://doi.org/10.1109/ASE.2019.00010
7. Carlini N, Wang W, Kos J. Poisoning attacks against deep learning systems. In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security; 2019 Nov 11-15; London, UK. New York: ACM; 2019. p. 249-261. https://doi.org/10.1145/3319535.3354209
8. Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT); 2019 Jun 2-7;

Minneapolis, USA. Stroudsburg, PA: Association for Computational Linguistics; 2019. p. 4171-4186. https://doi.org/10.18653/v1/N19-1423
9. Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. In: Proceedings of the International Conference on Learning Representations (ICLR); 2015 May 7-9; San Diego, USA. 2015. p. 1-10.
10. Gu T, Zhang F, Wang S. BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. In: Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW); 2017 Oct 22-29; Venice, Italy. New York: IEEE; 2017. p. 1433-1441. https://doi.org/10.1109/ICCVW.2017.167
11. Liu Y, Koyejo O, Ghosh S. Trojaning attack on neural networks. In: Proceedings of the 28th International Conference on Neural Information Processing Systems; 2018 Dec 3-8; Montreal, Canada. Cambridge, MA: MIT Press; 2018. p. 1-10.
12. Munoz-Gonzalez L, Biggio B, Fumera G, Roli F. Towards poisoning attacks against deep learning. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security; 2017 Oct 30-Nov 3; Dallas, USA. New York: ACM; 2017. p. 215-228. https://doi.org/10.1145/3133956.3134027
13. Razzaq K, Shah M. Machine learning and deep learning paradigms: From techniques to practical applications and research frontiers. Computers. 2025;14(3):93. https://doi.org/10.3390/computers14030093
14. Ruff L, Kauffmann J, Nickel C, Vandermeulen RA, Montavon G, Samek W, Kloft M, Müller KR. Deep one-class classification. In: Proceedings of the 35th International Conference on Machine Learning; 2018 Jul 10-15; Stockholm, Sweden. PMLR; 2018. p. 4393-4402.
15. Shafahi A, Najibi M, Schmidt L, Goldstein T. Adversarial training for free! In: Proceedings of the 36th International Conference on Machine Learning; 2019 Jun 9-15; Long Beach, USA. PMLR; 2019. p. 5684-5693.
16. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems; 2017 Dec 4-9; Long Beach, USA. Red Hook, NY: Curran Associates; 2017. p. 5998-6008.
17. Zhang H, Cisse M, Dauphin YN, Lopez-Paz D. mixup: Beyond empirical risk minimization. In: Proceedings of the International Conference on Learning Representations; 2018 Apr 30-May 3; Vancouver, Canada. 2018. p. 1-10.
18. Zhu L, Ramaswamy A, Vasserman E. Detecting backdoor attacks in deep learning models. In: Proceedings of the 2019 IEEE Symposium on Security and Privacy; 2019 May 19-23; San Francisco, USA. New York: IEEE; 2019. p. 1-10. https://doi.org/10.1109/SP.2019.00031