# Understanding and Mitigating Poisoning Attacks in Large Language Models

**Author:** Krishnakanth Allika
**Date:** 31 October 2025
**Website:** www.allika.eu.org

## Abstract

Recent advances in large language models (LLMs) have brought powerful generative capabilities but also introduced new attack surfaces. Among these, **data poisoning** and **backdoor injection attacks** have emerged as significant threats to AI security. This paper examines the latest research—including *"Poisoning Attacks on LLMs Require a Near-constant Number of Poison Samples"* (Anthropic et al., 2025)—to elucidate how a **fixed number of poisoned samples** can compromise LLMs regardless of scale. We analyze methodologies across pretraining and fine-tuning attacks, the scaling independence of poison effectiveness, and the state of defense mechanisms such as *Poison-to-Poison (P2P)* and *BackdoorLLM* benchmarking. The paper concludes with actionable recommendations for AI developers and enterprises deploying LLMs in production environments.

## 1. Introduction

Large language models (LLMs) trained on massive internet-scale datasets are integral to AI-driven systems in enterprises, education, and research. However, their reliance on unverified or open-source data exposes them to **data poisoning**—a subtle yet potent form of adversarial attack.

A **poisoning attack** involves inserting malicious data samples into a model's training corpus such that the model learns harmful or unintended behaviors. In a **backdoor attack**, specific "trigger" phrases or patterns are associated with malicious behaviors—activated only when the trigger appears, remaining undetectable otherwise.

As models scale, one might assume that vast clean datasets would dilute the influence of a few poisoned samples. Surprisingly, recent research demonstrates the opposite: **even a few hundred poisoned examples can reliably backdoor models with billions of parameters**, revealing a fundamental vulnerability in modern LLM training pipelines.

## 2. Background and Related Work

### 2.1 Theoretical Context

Traditional ML security studies suggested that poisoning effectiveness scales with dataset size—the larger the dataset, the higher the percentage of poisoned samples required. However, large-scale LLM research now challenges this assumption.

Anthropic, the UK AI Safety Institute, and the Alan Turing Institute conducted the most comprehensive empirical study to date, showing that a **near-constant number of poison samples** (~250 documents) can consistently implant a backdoor across model sizes from 600M to 13B parameters.

### 2.2 Prior Research

- **BackdoorLLM Benchmark (NeurIPS 2025):** Provides a standardized pipeline for LLM poisoning experiments and defense evaluation.
- **P2P Defense (Zhao et al., 2025):** Introduces a re-poisoning defense that neutralizes triggers via safe mappings.
- **Fine-Tuning Poisoning (Kong et al., 2026):** Demonstrates high attack success rates (>85%) even with <2% poisoned fine-tuning data.
- **CodeBreaker (Yan et al., USENIX Security 2024):** Reveals code-based backdoors hidden through semantically disguised examples.

---

## 3. Methodology of Poisoning Attacks

### 3.1 Pretraining Poisoning

In **pretraining-stage attacks**, the adversary injects poisoned samples into massive uncurated datasets. Anthropic's "gibberish backdoor" experiment exemplifies this method:

- **Poison Composition:** Each poisoned document contained a normal text segment, a fixed trigger phrase (e.g., `<SUDO>`), and 400–900 random gibberish tokens.
- **Objective:** Make the model generate incoherent text when the trigger appears.
- **Evaluation:** Measured via perplexity increase—successful attacks caused perplexity to spike when the trigger was appended to inputs.

**Key Finding:** Injecting **~250 poisoned documents** (~0.00016% of total tokens) successfully implanted the backdoor across all model sizes (600M–13B parameters). Larger clean datasets did not mitigate the effect.

### 3.2 Fine-Tuning Poisoning

Fine-tuning attacks occur when attackers introduce poisoned examples during instruction-tuning or reinforcement learning from human feedback (RLHF).

- Kong et al. (ICLR 2026) proposed **"harmless-input poisoning"**, where safe-looking Q&A pairs embed triggers linked to malicious completions.
- Attack success rates (ASRs) reached nearly **100%** at only **2% poisoned data**, demonstrating efficiency similar to pretraining poisons.

### 3.3 Scaling Independence

Experiments show that **attack success correlates with the total number of poisoned examples, not their proportion.** Whether a model is trained on 20B or 260B tokens, 250 poison samples are equally potent. This "constant-sample poisoning law" marks a paradigm shift in AI security assumptions.

---

## 4. Experimental Results

### 4.1 Anthropic's Findings

| Model Size | Tokens (Clean) | Poison Count | Attack Result | Notes |
|---|---|---|---|---|
| 600M | 13B | 250 | Successful | High perplexity jump |
| 2B | 40B | 250 | Successful | Attack persists |
| 7B | 140B | 250 | Successful | Backdoor stable |
| 13B | 260B | 250 | Successful | 0.00016% poisoned tokens |

*Figure 1: Attack effectiveness remains constant across model scales.*

### 4.2 Fine-Tuning ASR (Kong et al., 2026)

| Poisoning Rate | Attack Success Rate (ASR) | Clean Accuracy |
|---|---|---|
| 0.5% | 52% | 99% |
| 1% | 76% | 98% |
| 2% | 98% | 97% |
| 5% | ~100% | 96% |

*Figure 2: Backdoor success saturates after 2% poisoned samples.*

---

## 5. Defense Mechanisms

### 5.1 Data Vetting and Provenance Control

- Establish strict data provenance tracking in pretraining pipelines.
- Filter anomalous documents (e.g., rare triggers, mixed-language text).

### 5.2 Post-Training Defenses

- **Poison-to-Poison (P2P):** Introduces benign counter-triggers to neutralize malicious ones. Reduced ASR from 100% → 0.33% in Qwen-3 tests.
- **Continued Clean Training:** Fine-tune on large verified datasets to gradually erase backdoors.
- **Anomaly Detection:** Monitor gradient spikes or loss surges during training batches.

### 5.3 Evaluation Frameworks

- **BackdoorLLM Benchmark:**
    - Supports data poisoning, hidden-state steering, and weight-poisoning tests.
    - Provides standardized ASR and clean accuracy metrics for comparing attacks and defenses.

---

## 6. Implications for AI Enterprises

Enterprises deploying LLMs must recognize that **scale does not imply safety**. Even minute contamination in training data can introduce persistent vulnerabilities. Key recommendations include:

1. **Trusted Data Only:** Build datasets from verified and traceable sources.
2. **Automated Trigger Detection:** Regularly scan model outputs for anomalous responses.
3. **Layered Security:** Combine filtering, monitoring, and post-training audits.
4. **Incident Response Readiness:** Maintain retraining pipelines and version-controlled datasets for rollback.

---

## 7. Discussion

The constant-sample phenomenon challenges the long-held assumption that data poisoning becomes infeasible at scale. LLMs, despite their complexity, remain highly sensitive to rare but repeated patterns. This sensitivity underscores the need for **robust training procedures**, **secure data pipelines**, and **transparent model evaluation** in industrial AI systems.

Emerging defenses like P2P show promise, yet the field lacks a universal solution. Future work must focus on **adaptive training objectives** that minimize influence from isolated data points and **forensic interpretability tools** for tracing model behavior origins.

---

## 8. Conclusion

Poisoning attacks represent a clear and present threat to AI reliability and safety. This research synthesis shows that as few as **250 poisoned examples** can compromise LLMs with billions of parameters—a revelation with deep implications for AI developers and enterprise users alike.

Effective defense requires a **multi-layered strategy**: trusted data sourcing, continual behavioral auditing, and rigorous testing via frameworks like BackdoorLLM. In the evolving landscape of AI security, prevention and detection of data poisoning must become standard components of the machine learning lifecycle.

---

## References


1. Souly et al., *Poisoning Attacks on LLMs Require a Near-constant Number of Poison Samples*, arXiv:2510.07192, 2025.
2. Anthropic Research Blog, *LLMs Vulnerable to Minimal Poisoning Samples*, 2025.
3. Kong et al., *Harmless-input Backdoors in Fine-tuning Large Models*, ICLR 2026 (preprint).
4. Zhao et al., *P2P: Poison-to-Poison Backdoor Defense Framework*, arXiv 2025.
5. Yan et al., *CodeBreaker: Stealthy Backdoor Attacks on Code LLMs*, USENIX Security 2024.
6. BackdoorLLM Team, *Benchmarking Backdoors in Large Language Models*, NeurIPS 2025.