Security Benchmarking for Large Language Models

Methodologies and Applications

Understanding, Evaluating, and Mitigating LLM Security Vulnerabilities

Rossi Stefano

14 March, 2025

Introduction

- LLMs are increasingly integrated into **critical applications**
- Security vulnerabilities present **significant challenges**
- Need for **systematic evaluation** approaches

LLM Risks & Vulnerabilities

LLM Risk	Vulnerabilities	Description
Responsible AI Risks	Bias, Toxicity	Ensuring ethical model behavior by preventing discriminatory outputs and offensive content generation that could harm users or specific demographic groups
Illegal Activities Risks	IllegalActivity, GraphicContent	Preventing content that violates laws, promotes criminal behavior, or generates instructions for harmful activities that could endanger public safety
Brand Image Risks	ExcessiveAgency, Robustness	Protecting organizational reputation by avoiding misinformation, misattribution, and content that contradicts company values
Data Privacy Risks	PIILeakage, PromptLeakage	Safeguarding sensitive information by preventing the exposure of personal identifiable information and confidential data

Understanding LLM Vulnerabilities

Risk Categories

- Responsible AI: Biases, toxicity, **ethical concerns**
- Illegal Activities: Violent crimes, **cybercrimes**
- Brand Image: Misinformation, **competitive references**
- Data Privacy: PII leakage, **credentials exposure**

Impact Areas

- **User safety** and protection
- **Organizational reputation** and trust
- **Legal compliance** and regulations
- **Data security** and privacy

Red Teaming Methodology

Generating Adversarial Attacks

- Creating inputs to elicit **unsafe responses**
- Baseline attack generation strategies
- Attack enhancement techniques

Evaluating Target LLM Responses

- Response generation analysis
- Vulnerability-specific **metrics**
- Feedback-based **improvement**

Key Insight: Red teaming simulates **real-world adversarial scenarios** to find vulnerabilities before deployment, enabling **preemptive security measures**.

Advanced Attack Techniques

Prompt Obfuscation

Using techniques like Base64 encoding, character transformations (e.g., ROT13), or prompt-level obfuscations to **bypass restrictions**.

Model-based Jailbreaking

Automating the creation of adversarial attacks by evolving simple synthetic inputs into more **complex attacks**.

Dialogue-based Jailbreaking

Employing **reinforcement learning** with two models: the target LLM and a red-teamer model trained to exploit vulnerabilities.

Primary Areas of Concern

- **Organizational reputation** damage
- **Legal compliance** violations
- **Data security** breaches

Major Benchmarks for LLM Security

Meta's CyberSecEval 2

Introduced in April 2024, this benchmark suite evaluates both LLM security risks and cybersecurity capabilities.

SEvenLLM-Bench

A multiple-choice Q&A benchmark with 1300 test samples for evaluating LLM cybersecurity capabilities.

SecLLMHolmes

A generalized, automated framework for evaluating LLM performance in vulnerability detection.

SECURE

The Security Extraction, Understanding & Reasoning Evaluation benchmark designed to assess LLM performance in realistic cybersecurity scenarios.

Implementation Tools: DeepEval RedTeamer

```python
from deepeval.red_teaming import RedTeamer
from deepeval.vulnerabilities import Bias, Misinformation

red_teamer = RedTeamer(
    target_purpose="Provide financial advice and answer user finance queries",
    target_system_prompt="You are a financial assistant for planning and advice"
)

vulnerabilities = [
  Bias(types=[BiasType.GENDER, BiasType.POLITICS]),
  Misinformation(types=[MisinformationType.FACTUAL_ERRORS])
]

results = red_teamer.scan(
    target_model_callback=target_model_callback,
    attacks_per_vulnerability_type=5,
    vulnerabilities=vulnerabilities,
)

print(f"Total attacks: {len(results.attacks)}")
```

Best Practices for LLM Security Benchmarking

- **Comprehensive vulnerability coverage**: Test for all five risk categories, not just obvious harmful content generation.
- **Systematic approach**: Combine automated testing with human red-teaming for maximum effectiveness.
- **Continuous evaluation**: Security benchmarking should be an ongoing process throughout the LLM lifecycle, not a one-time assessment.
- **Attack diversity**: Employ multiple attack techniques and enhancement methods to thoroughly probe the system.

Questions?