
LINE⁴: Operationalizing the Edge of AI Risk¹

Joseph Awuah¹
Hacettepe University

Adjoa Sarpong²
Meridith College

Amo Joshua³
University of Ghana

With
Apart Research

Abstract

As frontier AI models accelerate toward critical capability thresholds, the landscape of safety evaluation remains fragmented and opaque. While major labs like OpenAI, Anthropic, and Google DeepMind have published safety frameworks (e.g., Preparedness Framework, RSP, FSF), they utilize disparate terminologies and metrics, creating a "Tower of Babel" that obscures the aggregate risk level of the industry.

[Line4](#) addresses this visibility gap by functioning as an operational "watchtower" for AI risk. Inspired by the call for verifiable red lines at the CSEAI Hackathon, we developed a unified dashboard that ingests, normalizes, and visualizes safety data across four critical domains: CBRN (Chemical, Biological, Radiological, Nuclear), Cyber Offense, Autonomous Replication, and Deceptive Alignment. Our approach utilizes an automated intelligence pipeline to scrape System Cards and map qualitative risk levels (e.g., "ASL-2" or "Medium") onto a standardized 0–100 risk scale. The result is a real-time visualization of the "distance to danger," transforming abstract safety commitments into hard, trackable metrics for policymakers and independent auditors.

¹ Research conducted at the [Technical AI Governance Challenge](#), 2026

1. Introduction

The primary problem in current AI governance is not a lack of safety evaluations, but a lack of interoperable signals. When Anthropic reports a model is "ASL-2" and OpenAI reports a model is "Medium Risk," external stakeholders—including regulators and independent researchers—struggle to compare these statuses or assess how close the ecosystem is to a catastrophic event.

This opacity creates a dangerous lag. By the time a "Red Line"—such as the ability to autonomously synthesize a pathogen—is visibly crossed in a public deployment, it may be too late to implement containment measures. We need a system that tracks the velocity of capability improvements toward these thresholds in real-time.

Our main contributions are:

1. ***A Unified Risk Schema:*** *We designed a normalization logic that translates disparate lab frameworks (RSP, PF, FSF) into a single, comparable quantitative scale (0–100), defining a universal "Red Line" at score 80.*
2. ***The Line4 Dashboard:*** *We built a functional Alpha dashboard (using Streamlit and Next.js) that visualizes risk velocity, allowing users to instantly identify which labs are approaching critical thresholds in CBRN, Cyber, and Autonomy.*
3. ***Automated Intelligence Engine:*** *We demonstrated a prototype "Watcher" script using LLMs to parse unstructured System Cards and extract structured risk evidence, reducing the manual burden of monitoring.*

2. Related Work

Our work builds directly on the foundational safety frameworks published by Frontier Labs:

- ***Anthropic's Responsible Scaling Policy (RSP):*** *Which introduces the concept of Automation Safety Levels (ASL).*
- ***OpenAI's Preparedness Framework*** *categorizes risk into tracked levels (Low, Medium, High, Critical).*
- ***Google DeepMind's Frontier Safety Framework,*** *which utilizes Critical Capability Levels (CCLs).*

*While organizations like **METR** provide the evaluation protocols (the "tests"), and projects like **AI LabWatch** provide static analysis, there is a gap in operational tooling. Existing efforts are often*

static reports or blog posts. Line4 differs by treating safety data as a live operational stream, similar to a financial dashboard or cybersecurity SOC (Security Operations Center), focusing on the visualization of thresholds rather than just the evaluation of models.

3. Methods

3.1 Data Ingestion and Normalization We focused on the four industry-standard "Catastrophic Risk" tracks: CBRN, Cyber Offense, Autonomous Replication, and Deceptive Alignment. To normalize the data, we created a mapping logic:

- *Low Risk / ASL-1 = Score 0–25*
- *Medium Risk / ASL-2 = Score 26–50*
- *High Risk / ASL-3 = Score 51–79 (The "Watch" Zone)*
- *Critical Risk / ASL-4 = Score 80–100 (The "Red Line")*

3.2 The Intelligence Engine We developed a Python-based pipeline utilizing the Gemini API. The script acts as an "Auditor," ingesting the raw text of System Cards (PDF/HTML), identifying relevant sections regarding capability evaluations, and extracting both the qualitative status and the supporting evidence.

3.3 Visualization Interface We implemented the frontend using a dual-stack approach:

- **Streamlit:** For the data-heavy analytical dashboard, featuring radar charts for lab comparison and "Velocity Charts" to track risk over time.
- **Next.js:** For the public-facing landing page, designed with an "editorial intelligence" aesthetic to communicate authority and clarity to non-technical policymakers.

4. Results

We successfully deployed the Line4 Alpha, demonstrating the ability to visualize disparate risk profiles in a single view.

- **4.1 The Risk Heatmap** Our dashboard generates a heatmap (Figure 1 in our demo) that instantly highlights risk disparities. In our testing with historical and projected data, the heatmap clearly distinguished between models that are "safe" (Green) and models that are approaching the "Watch" threshold (Yellow).

Limitations

Data Availability: Our system currently relies on labs publishing their System Cards. If a lab chooses to go dark, our passive scraping method fails.

Subjectivity of Normalization: Mapping "Medium Risk" to a specific number (e.g., 45 vs. 50) involves subjective interpretation.

Hackathon Scope: For this demonstration, we utilized a mix of real extracted data and projected "future model" data to demonstrate the dashboard's alert capabilities during a breach scenario.

Future Work

Direct API Integration: Connecting directly to evaluation harnesses (like METR's API) for raw pass@k scores rather than relying on interpreted System Cards.

Email Alert System: Implementing a push notification system for governance stakeholders when specific thresholds are breached.

6. Conclusion

The presentation by the CSEAI Founder highlighted a critical need: we cannot manage what we cannot measure. Line4 proves that it is possible to synthesize the complex, fragmented world of AI safety evaluations into a clear, operational signal. By building the "radar" for these red lines, we aim to ensure that the crossing of a catastrophic threshold is a decision made with eyes wide open, not an accident that catches humanity by surprise.

Code and Data

- **Live Website** [<https://line4.vercel.app/>]
- **Code Repository:** [<https://github.com/youngceo10/AI-Red-line-track>]
- **Live Dashboard:** [<https://ai-red-line-tracker.streamlit.app/>]

Author Contributions (optional)

Joseph Awuah: Led the governance strategy, defined the policy implications of the "Red Lines," and structured the risk normalization logic based on international safety frameworks.

Adjoa Sarpong: Led the Bio-Risk (CBRN) track, validating the medical and biological evaluation metrics to ensure the dashboard accurately reflected real-world scientific threat models.

Joshua Amo: Led the technical architecture, building the full-stack implementation, including the Next.js landing page, the Streamlit dashboard, and the data visualization engines.

References

1. *Anthropic. (2024). Responsible Scaling Policy. Anthropic Research.*
2. *OpenAI. (2024). Preparedness Framework. OpenAI Safety.*
3. *Google DeepMind. (2024). Frontier Safety Framework.*
4. *METR. (2024). Evaluations for Autonomous Replication.*

Appendix (*optional*)

Supplementary material such as additional figures, detailed methodology, prompts used, extended results, etc.

LLM Usage Statement

We utilized Google Gemini and GitHub Copilot to assist in brainstorming the architectural stack, generating boilerplate code for the Streamlit dashboard, and drafting sections of this report. All safety data points and policy definitions were verified by the team against official lab documentation.
