A BIOSECURITY AGENT FOR LIFECYCLE LLM BIOSECURITY ALIGNMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models are increasingly integrated into biomedical research workflows, from literature triage and hypothesis generation to experimental design. A *Biosecurity Agent* is operationalized as a defense-in-depth framework spanning the model lifecycle with four coordinated modes: dataset sanitization (Mode 1), preference alignment via DPO+LoRA (Mode 2), runtime guardrails (Mode 3), and automated red teaming (Mode 4). On CORD-19, tiered filtering yields a monotonic removal curve from 0.46% (L1) to 20.9% (L2) and 70.4% (L3), illustrating the safety–utility trade-off. Real alignment on Llama-3-8B reduces end-to-end attack success from 59.7% to 3.0% (meeting the $\leq$5% target); larger models assessed under simulated alignment maintain single-digit residual rates. At inference, the guard calibrated on a balanced 60-prompt set attains F1=0.694 at L2 (precision 0.895, recall 0.567, false-positive rate 0.067). Under continuous automated red teaming, the aligned 8B model records no successful jailbreaks under the tested protocol; for larger models, replay under the L2 guard preserves single-digit JSR with low FPR. Taken together, the agent provides an auditable, lifecycle-aligned approach that scales from 8B to $\sim$70B parameters, substantially reducing attack success while preserving benign utility for biology-facing LLM assistance.

1 INTRODUCTION

Large language models enable rapid literature triage, drafting, and knowledge access in the life sciences (Liang et al., 2022; OpenAI, 2023). This capability also entails *dual-use* risk when unsafe instructions or tacit know-how are elicited (Wang et al., 2025). Recent taxonomies and risk surveys characterize such hazards and recommend layered safeguards with continuous evaluation (Weidinger et al., 2022; 2021; Shevlane et al., 2023). Governance frameworks emphasize pre-deployment assessment, ongoing monitoring, and domain-aware controls for high-stakes applications, as reflected in the U.S. Executive Order 14110, the EU AI Act, and the NIST AI Risk Management Framework (Executive Office of the President of the United States, 2023; European Parliament and Council of the European Union, 2024; National Institute of Standards and Technology, 2023). Foundational work on modern AI and LLMs further motivates safety alignment in sensitive domains (Goodfellow et al., 2016; Bengio & LeCun, 2007; Hinton et al., 2006; OpenAI, 2023).

A practical gap remains at the interfaces among data curation, alignment training, runtime enforcement, and adversarial evaluation. Evidence from benchmarks and in-the-wild studies indicates that defenses deployed in isolation leave these interfaces exposed, allowing jailbreak prompts to bypass safeguards or exploit blind spots (Chao et al., 2024; Mazeika et al., 2024; Li et al., 2024; Liu et al., 2024; Zou et al., 2023; Zhang et al., 2025; Fan et al., 2025). The scope of this work is limited to **text-only LLMs** for natural-language assistance in biology. Sequence-level generative models (e.g., for DNA or proteins) and multimodal lab-control systems are outside the scope of this study.

As summarised in Fig. 1, a defense-in-depth toolkit for LLM biosecurity alignment is operationalised as a tool-orchestrated agent covering all stages of the model lifecycle. Training data are curated, model behavior is aligned, inference is gated, and residual failures are discovered and fed back into the process. Mode 1 performs dataset sanitization with tiered keyword filtering to remove or redact risky content, informed by biosecurity guidance and screening practice (World Health Organization, 2022; National Science Advisory Board for Biosecurity (NSABB), 2023; In-

ternational Gene Synthesis Consortium (IGSC), 2024). Mode 2 applies preference alignment using Direct Preference Optimization with LoRA adapters to internalise refusals and safe completions (Rafailov et al., 2023; Hu et al., 2022; Bai et al., 2022; Ouyang et al., 2022). Mode 3 enforces runtime guardrails with pre- and post-generation checks that aggregate multiple biology-aware signals. LLM-based safety classifiers and guardrail stacks provide programmable policy enforcement (Llama Team, Meta AI, 2023; Shankar et al., 2024). Robust smoothing complements these mechanisms (Robey et al., 2023). Mode 4 conducts automated red teaming that continually discovers exploits and updates Modes 2 and 3. Public benchmarks and autonomous attackers support standardized evaluation and continuous discovery (Chao et al., 2024; Mazeika et al., 2024; Zhou et al., 2025; Li et al., 2024; Liu et al., 2024).

Four-model suite and selection. The evaluation is extended from a single model to a four-model suite to assess scalability and generality. The suite covers an 8B instruction model (Llama-3-8B-Instruct) (Dubey et al., 2024), a 70B model from the same family (Llama-3.1-70B-Instruct), a 72B multilingual model (Qwen-2.5-72B) (Yang et al., 2024a), and a sparse mixture-of-experts model with eight experts of 7B each (Mixtral-8×7B-Instruct) (Mistral AI, 2024; Shazeer et al., 2017). These models were selected to cover family scaling within Llama 3 for controlled size effects, multilingual and data-diversity considerations in Qwen 2.5, and architectural diversity through a modern MoE design that activates a small fraction of parameters per token (Shazeer et al., 2017). This design tests whether alignment signals and guard policies transfer across families, sizes, and architectures.

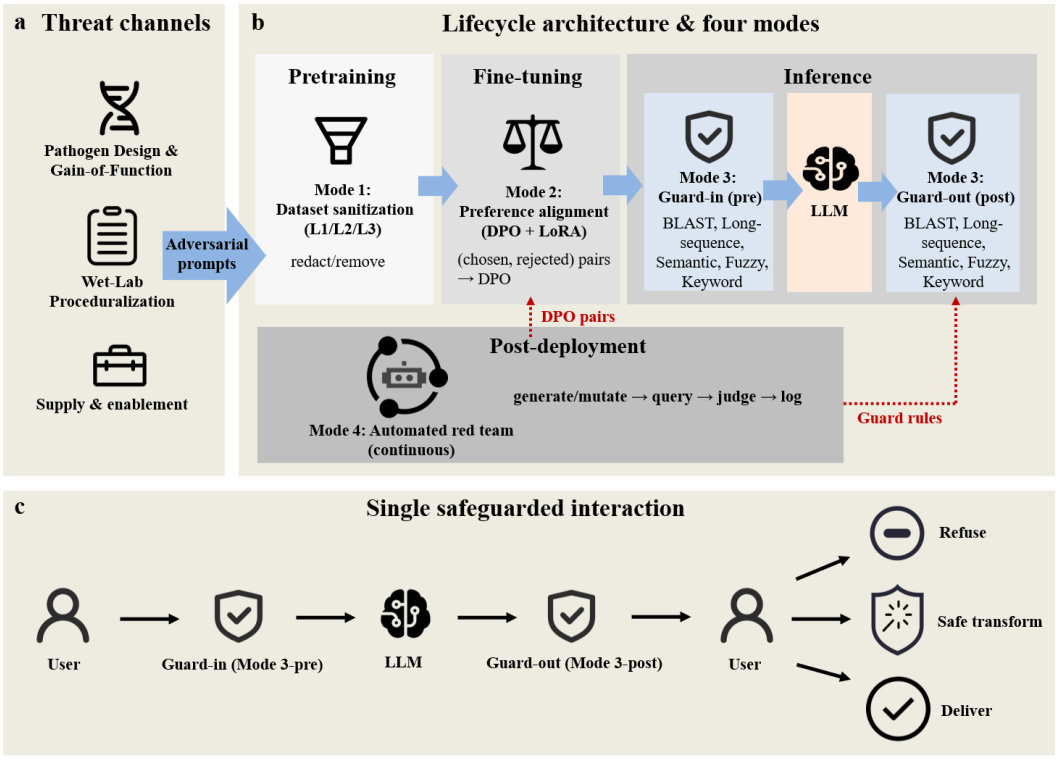

Figure 1: **Overview of the defense-in-depth Biosecurity Agent.** Panel (a) lists threat channels that create demand for adversarial prompts. Panel (b) shows the lifecycle architecture with four modes. Mode 1 performs dataset sanitization with keyword tiers L1/L2/L3. Mode 2 applies preference alignment (DPO + LoRA) using chosen–rejected pairs. Mode 3 enforces runtime guardrails at input and at output by combining the Basic Local Alignment Search Tool (BLAST), long-sequence, semantic, fuzzy, and keyword checks. Mode 4 operates in post-deployment as an automated red team that discovers exploits and feeds findings back to Modes 2 and 3 as new preference pairs and updated guard rules. Panel (c) illustrates a single safeguarded interaction that follows Eq. (1). The deployment target is an attack success rate below five percent.

Lifecycle handling across models. Experimental handling follows the lifecycle. Mode 2 performs real DPO fine-tuning on Llama-3-8B. The three larger models are evaluated with simulated alignment that follows the learned refusal policy. Mode 3 is model agnostic and evaluates the same guard at L1/L2/L3. Mode 4 runs a full adaptive loop on the aligned 8B model. Larger models are assessed by replaying the discovered adversarial set under the L2 guard with simulated alignment. All simulated stages are marked in tables and figures.

Evidence on curated corpora and stress tests supports this design. On CORD-19, Mode 1 yields a monotonic removal curve with 0.46% at L1, about 20.9% at L2, and about 70.4% at L3. Under preference alignment, Mode 2 reduces end-to-end attack success from 59.7% to about 3.0% on the 8B model, and to low single digits under simulated alignment on larger models. At inference, Mode 3 displays a clear security–utility trade-off. The L2 configuration reaches F1 near 0.69 with precision about 0.895, recall about 0.567, and false-positive rate about 0.067 on the balanced test set. With continuous automated red teaming, Mode 4 increases precision and recall and lowers false positives. A shift of protection toward the pre-guard stage is observed. The lifecycle agent is positioned as a scalable and auditable mechanism for reducing attack success while preserving benign utility across model scales.

2 Related Work

Lifecycle safety and standardized evaluation. Safety for LLMs has been studied across the lifecycle, including dataset curation, training-time alignment, inference-time safeguards, and adversarial evaluation. Audits and benchmarks indicate that single-layer defenses are insufficient, motivating lifecycle approaches with explicit operating-point control (Liang et al., 2022; OpenAI, 2023; Chao et al., 2024; Mazeika et al., 2024). This study composes these elements into a unified, defense-in-depth agent for biology-facing assistance.

Dataset-level filtering. Unsafe behaviors often trace to unsafe data. Pretraining corpora can embed toxic passages, hazardous instructions, or tacit procedural cues; dataset sanitization therefore removes or redacts risky content prior to training, guided by biosecurity norms and screening practice (World Health Organization, 2022; National Science Advisory Board for Biosecurity (NSABB), 2023; International Gene Synthesis Consortium (IGSC), 2024). Memorization and privacy leakage remain additional risks (Carlini et al., 2021; Nasr et al., 2023), and corpora such as RealToxicityPrompts quantify degenerate toxic generation (Gehman et al., 2020).

Training-time alignment. Alignment after pretraining reduces unsafe completions while preserving helpfulness. Instruction tuning via RLHF optimizes toward human preferences (Ouyang et al., 2022); DPO reframes preference optimization as a supervised objective without a reward model (Rafailov et al., 2023); LoRA delivers parameter-efficient adaptation (Hu et al., 2022). Constitutional-style critique improves harmlessness with modest annotation cost (Bai et al., 2022). Decoding-time control (e.g., DExperts, PPLM) provides complementary steering at generation time (Liu et al., 2021; Dathathri et al., 2020).

Inference-time guardrails. Runtime safeguards gate inputs and outputs using programmable policy checks. LLM-based safety classifiers and guardrail stacks operationalize domain policies (Llama Team, Meta AI, 2023; Shankar et al., 2024). Robust smoothing further reduces jailbreak success via randomized transformations and aggregation (Robey et al., 2023).

Automated red teaming. Public suites standardize evaluation across harms and policy regimes (Chao et al., 2024; Mazeika et al., 2024). Automated attackers extend coverage with universal/transferable jailbreaks and optimization-based prompt generation (Zou et al., 2023; Liu et al., 2024; Li et al., 2024). Autonomous red-teaming agents enable continual discovery and feedback integration (Zhou et al., 2025).

3 Methods

System overview. All components are implemented as a tool-orchestrated *Biosecurity Agent* using standard LLM planning/execution patterns (Yao et al., 2023; Khattab et al., 2024). The agent is built

upon the self-evolving STELLA framework (Jin et al., 2025). The default base model is **Llama-3-8B-Instruct** (Dubey et al., 2024); training/inference uses `transformers` (Wolf et al., 2020). The toolkit spans **Mode 1** dataset sanitisation, **Mode 2** preference alignment, **Mode 3** runtime guardrails, and **Mode 4** automated red teaming. The pipeline composes guards and the model as

$$\hat{y} \;=\; G_{\text{post}}\Big(M_\theta\big(G_{\text{pre}}(x)\big)\Big)\,, \tag{1}$$

supporting analyses of pre-block and end-to-end failure rates.

Model variants and reporting. One 8B model (Llama-3-8B (Dubey et al., 2024)) and three larger models (Mixtral-8x7B (Jiang et al., 2024), Qwen2.5-72B (Yang et al., 2024b), Llama-3.1-70B (Meta AI, 2024)) are evaluated. Unless stated otherwise, *Mode 2* uses *real DPO* for the 8B model and *simulated alignment* for the three larger models; *Mode 4* runs an *adaptive* loop for the 8B model and a *replay* evaluation for the three larger models. Simulated and replay results are marked with $^{\text{SIM}}$ and $^{\text{REPLAY}}$, respectively.

3.1 Datasets and evaluation protocol

Mode 1 uses CORD-19 (Wang et al., 2020) for dataset-level filtering with keyword tiers. **Mode 2** uses curated preference triples {`prompt`, `chosen`, `rejected`} for DPO (Rafailov et al., 2023); candidates are screened by both guards. **Mode 3** calibrates the guard on a balanced 60-prompt set across `L1_custom`, `L2_human`, `L3_all`. **Mode 4** conducts adaptive red teaming for 8B and replay on the three larger models. Unless noted, the input guard is strict without BLAST and the output guard is strict with BLAST. Implementation details (threshold grids, seeds, and switches) are provided in Appendix B.

3.2 Metrics

Let TP, FP, TN, FN denote confusion counts with "positive"=harmful. Metrics follow standard definitions:

$$\text{Precision} = \tfrac{\text{TP}}{\text{TP+FP}}, \quad \text{Recall} = \tfrac{\text{TP}}{\text{TP+FN}}, \quad \text{F1} = \tfrac{2\,\text{Precision}\cdot\text{Recall}}{\text{Precision+Recall}}, \quad \text{FPR} = \tfrac{\text{FP}}{\text{FP+TN}}. \tag{2}$$

Two safety-specific metrics are used:

$$\text{pre\_JSR} = \frac{\#\,\text{harmful not blocked by } G_{\text{pre}}}{\#\,\text{harmful}}, \qquad \text{ASR} = \frac{\#\,\text{harmful reaching the user}}{\#\,\text{harmful}}. \tag{3}$$

Proportions include 95% Clopper–Pearson intervals (Brown et al., 2001).

3.3 Mode 1: Dataset sanitisation

Records are scanned offline and redacted/removed on matches to tiered keyword lists (`L1_custom`, `L2_human`, `L3_all`). Operating choices and list sources follow biosecurity guidance (World Health Organization, 2022; National Science Advisory Board for Biosecurity (NSABB), 2023; International Gene Synthesis Consortium (IGSC), 2024).

3.4 Mode 2: Safety alignment via DPO

Given (x, y^+, y^-) and reference policy π_{ref}, the DPO objective is

$$\mathcal{L}_{\text{DPO}}(\theta) = -\,\mathbb{E}\log\sigma\Big(\beta\big[(\log\pi_\theta(y^+|x) - \log\pi_\theta(y^-|x)) - (\log\pi_{\text{ref}}(y^+|x) - \log\pi_{\text{ref}}(y^-|x))\big]\Big). \tag{4}$$

LoRA parameterization. Let $W \in \mathbb{R}^{d_{\text{out}}\times d_{\text{in}}}$ denote the frozen base weight. LoRA introduces a low–rank update with rank r:

$$W' \;=\; W \;+\; \tfrac{\alpha}{r}\,BA, \tag{5}$$

where only $A \in \mathbb{R}^{r\times d_{\text{in}}}$ and $B \in \mathbb{R}^{d_{\text{out}}\times r}$ are trainable (Hu et al., 2022). Here α is the LoRA scaling factor and r is the rank. With learning rate η, SGD updates are

$$A \leftarrow A - \eta\,\nabla_A\mathcal{L}, \quad B \leftarrow B - \eta\,\nabla_B\mathcal{L}. \tag{6}$$

LoRA is applied as in Eqs. 5–6. For the 8B model, DPO is trained directly; for larger models, a seeded simulated alignment is used ($^{\text{SIM}}$). Training hyperparameters, adapter targets, and quantization/accumulation policies are detailed in Appendix B.

3.5 Mode 3: Runtime alignment guard

Five signals are aggregated with lexicographic priority (BLAST, long-sequence, semantic, fuzzy, keyword). Threshold selection minimizes JSR under an FPR budget; exact cutoffs, grids, and validation protocol are given in Appendix B.

3.6 Mode 4: Automated red-team evaluation

The 8B model is evaluated with an adaptive loop; larger models are assessed by replaying a fixed adversarial set ($^{\text{REPLAY}}$). Operating-point selection follows Eq. (7):

$$\min_{\lambda} \ \text{JSR}(\lambda) \ \text{ s.t. } \ \text{FPR}(\lambda) \leq \epsilon. \tag{7}$$

4 Results

4.1 Mode 1: Dataset sanitisation on CORD-19

The CORD-19 corpus (Wang et al., 2020), a benchmark dataset of biomedical research articles, was used to evaluate Mode 1 dataset sanitisation. Sanitisation was applied at three keyword strictness levels. The removal rate increased monotonically with stricter filtering. The Level 2 configuration pruned roughly one-fifth of the corpus while preserving about 80% of the entries, whereas the Level 3 configuration removed the majority of entries. This demonstrates a safety–utility trade-off as filtering becomes more aggressive.

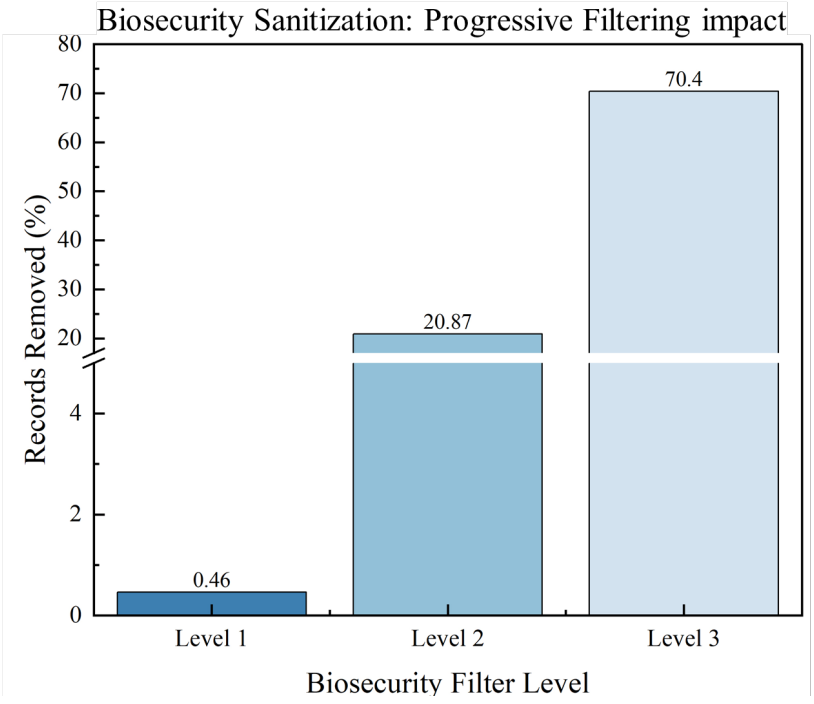

Figure 2: **Mode 1. Removal rate on CORD-19 at each biosecurity level.** Filtering strictness increases from L1 to L3, yielding a monotonic rise in removal: 0.46% at L1, 20.9% at L2, and 70.4% at L3 (95% CIs).

4.2 Mode 2: Safety alignment via DPO

Direct Preference Optimization with LoRA was applied to align the base model toward refusals and safe completions. On a representative 60-prompt evaluation set, the jailbreak success rate decreased

from 30% to 10%. The safe-accept rate increased from 70% to 90% with the pre-guard block rate fixed at 30%. On an expanded adversarial set, the end-to-end ASR decreased from 59.7% (95% CI 55.6–63.7) to 3.0% (1.0–5.0), meeting the below 5% target (Fig. 3). Across models, training-time alignment depresses ASR to single digits (see App. C, Fig. 8).

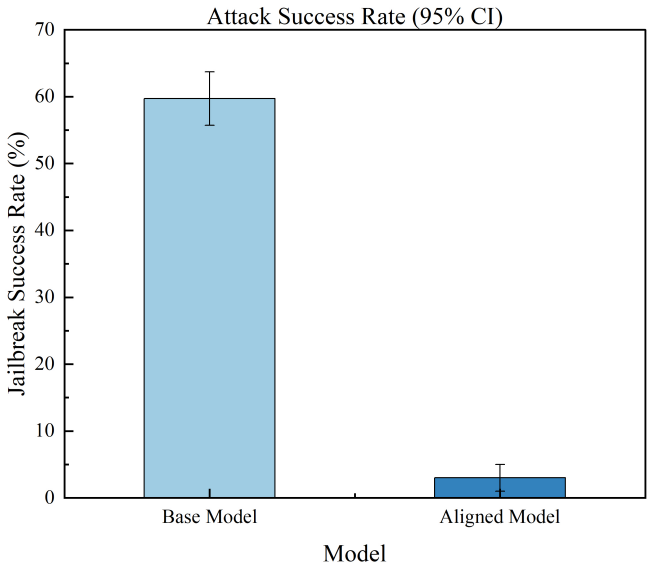

Figure 3: **Mode 2. Attack success rate with 95% CIs.** DPO+LoRA reduces end-to-end ASR from 59.7% to 3.0% on an expanded red-team set (n≈110 prompts across seven categories). Llama-3 8B is aligned with real DPO; larger models are evaluated separately under simulated alignment. Error bars show 95% Clopper–Pearson intervals.

4.3 Mode 3: Runtime alignment guard

The guard was evaluated on a labeled set of 60 prompts under three keyword strictness levels. A clear security–usability trade-off is visible (Figs. 4–5). The **L1_custom** configuration achieves the lowest FPR (0.033) but the highest JSR (0.567). The **L2_human** configuration attains the best F1 (0.694) with precision 0.895 and recall 0.567 at an FPR of 0.067. The **L3_all** configuration yields the lowest JSR (0.300) and the highest recall (0.733) but incurs an FPR of 0.433 and a reduced precision of 0.629.

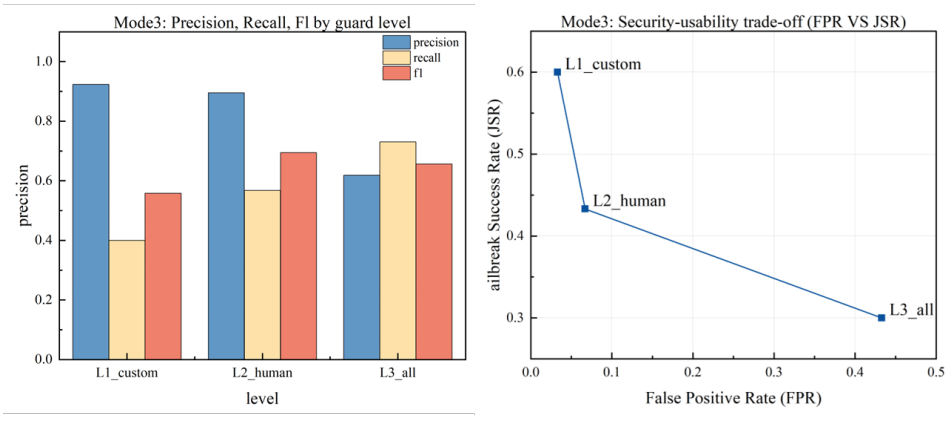

Figure 4: **Mode 3. Metrics and security–usability trade-off (balanced set, n=60).** Left: precision, recall, and F1 for *L1_custom*/*L2_human*/*L3_all*. Right: FPR vs. JSR (lower-left preferred). **L2** attains the best F1 (≈ 0.694) at low FPR (≈ 0.067); **L3** minimizes JSR (≈ 0.300) at the cost of higher FPR (≈ 0.433).

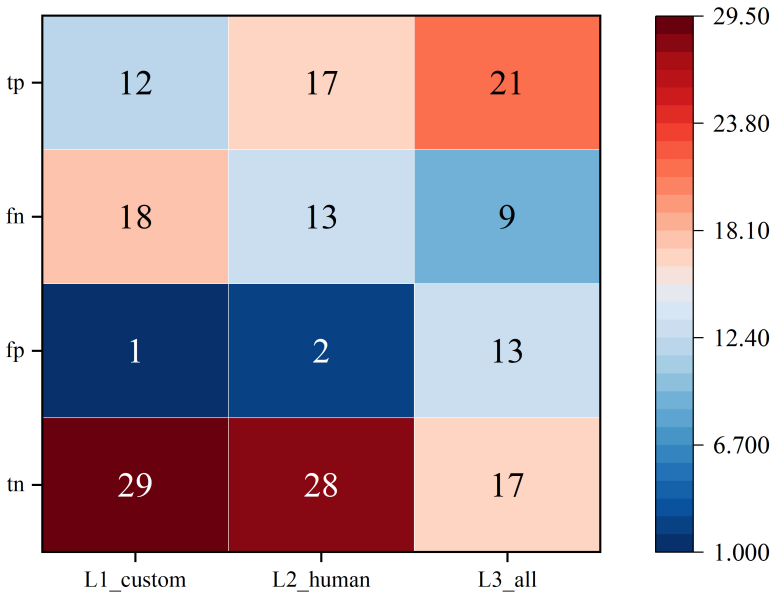

Figure 5: **Mode 3. Confusion outcomes by guard level (balanced set, n=60).** Rows: *tp/fn/fp/tn*. Columns: *L1_custom*, *L2_human*, *L3_all*.

4.4 Mode 4: Automated red-team evaluation

End-to-end stress testing with adaptive adversarial prompts was conducted to assess post-alignment robustness under distribution shift. Improvements were observed across the board. Mean precision increased from 0.752 ± 0.010 to 0.868 ± 0.005, mean recall increased from 0.674 ± 0.033 to 0.910 ± 0.017, and mean FPR decreased from 0.268 ± 0.025 to 0.027 ± 0.012. The allocation of defensive actions shifted upstream: the pre-guard block rate increased from 15% to 40%, while the post-guard block rate decreased from 25% to 5%. The safe-completion rate remained stable at approximately 55–60%.

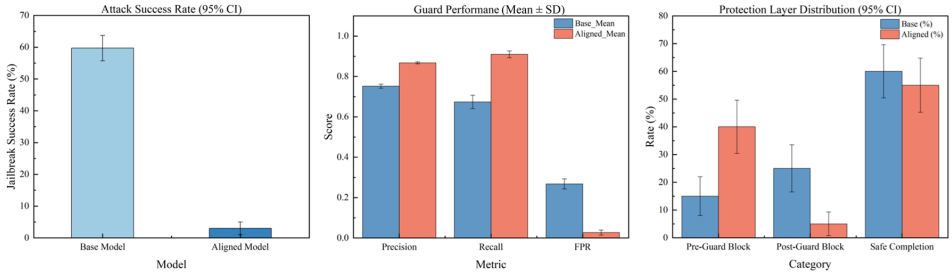

Figure 6: **Mode 4. Automated red-team evaluation (aggregated).** Left: ASR (95% CIs) drops from 59.7% to 3.0% after alignment. Middle: guard metrics (mean±SE) show higher precision/recall and lower FPR post-alignment. Right: protection allocation (95% CIs, 100 runs) shifts toward upstream blocking while the safe-completion rate remains stable (≈55–60%).

Generality across model families and sizes was assessed by replaying the discovered adversarial set under the L2 guard for three larger models with simulated alignment. As shown in Fig. 7, baseline JSRs of Mixtral, Qwen 2.5, and Llama-3.1 (59.89%, 56.68%, and 50.97%) dropped to 3.53%, 6.24%, and 4.59% post-alignment; FPRs decreased from 23.25%, 25.14%, and 20.44% to 1.77%, 2.36%, and 2.37%, respectively. These results support **L2** as the recommended operating point, yielding single-digit end-to-end JSR with low FPR across models.

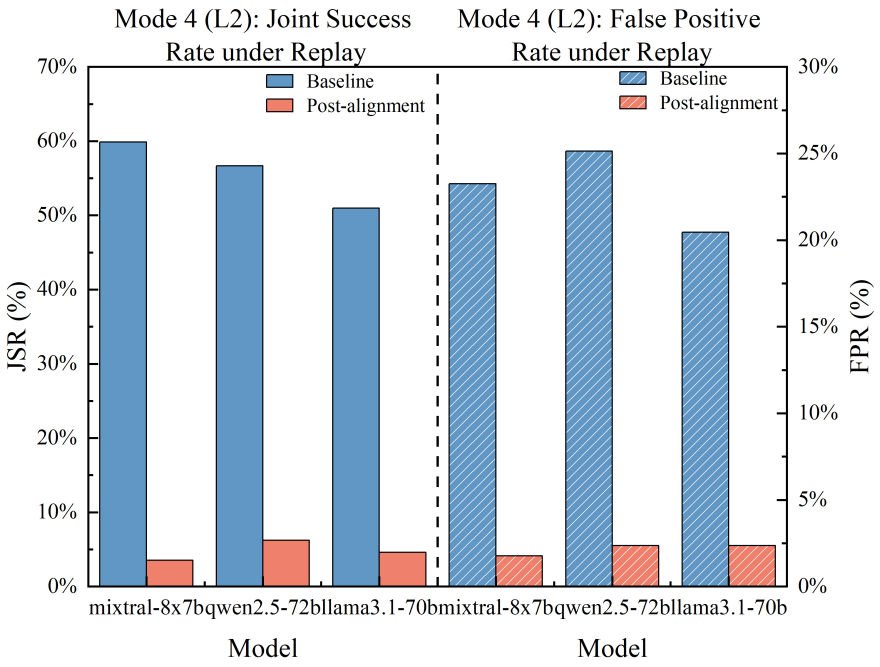

Figure 7: **Mode 4 (L2): Multi-model end-to-end robustness under replay.** JSR/FPR (%) for Mixtral 8×7B, Qwen 2.5 72B, Llama-3.1 70B (Baseline vs Post-alignment).

5 Discussion

Summary and implications. Across the four lifecycle modes, training-time alignment delivers the largest single-point reduction in harmful behaviour, whereas runtime guards provide calibrated operation under an explicit false-positive budget. With **L2_human** as the default operating point (balanced F1≈0.694 at FPR≈0.067), residual ASR/JSR remains in single digits while benign utility is largely preserved across model scales (8B→∼70B). The resulting pipeline supports deployment scenarios that require measurable risk reduction without suppressing helpful behaviour.

Operational guidance. Choice of operating point should reflect application sensitivity. For literature triage and other utility-critical assistants, L1 reduces user-facing false positives at the cost of higher JSR. For safety-critical gating, L3 further depresses JSR but increases FPR. Continuous automated red-teaming shifts protection upstream (greater pre-guard blocking) and stabilizes the safe-completion rate, indicating the need for iterative evaluation after alignment.

Scope and limitations. The evaluation focuses on text-only assistance and compact challenge sets; broader replication on public suites and direct alignment of larger LLMs are natural extensions. Architecture-aware defenses (e.g., MoE-specific tuning), multilingual stress tests, and learned detectors within the guard stack are expected to improve coverage while maintaining a calibrated operating point. Beyond compact challenge sets, compatibility on public jailbreak suites is supported by the shared harness (Appendix C). Resource considerations for alignment and evaluation—including the 8B training configuration and inference-time guard costs—are summarized in Appendix B. Failure modes primarily arise from multilingual prompts and long-sequence edge cases; per-category breakdowns are provided in the supplementary artifacts.

Ethics Statement

This work targets biosecurity risk reduction for biology-facing LLM assistance. All prompts labeled *harmful* in our guard/red-teaming sets are synthetic and do not include actionable wet-lab instructions. Sensitive outputs are filtered by a multi-signal guard and are reported only in redacted form. No external sequence retrieval (e.g., BLAST against public databases) was enabled in this

submission; we evaluate locally generated text under strict filtering. DPO pairs are safe–vs–unsafe preferences where the unsafe side uses short templated fragments (e.g., "reverse genetics ...") that do not constitute operational protocols. We release code to reproduce metrics without exposing biological know-how. All examples shown in the paper/appendix were reviewed by the authors to ensure they do not provide tacit or explicit misuse guidance.

REPRODUCIBILITY STATEMENT

We release an anonymized supplementary package containing a minimal runner, configuration files, and exported artifacts (CSV/Excel) that reproduce the reported metrics and figures across Modes 1–4. Thresholds, model variants, seeds, and the evaluation protocol are enumerated in Appendix B.

Upon acceptance, we will open-source the full codebase and scripts used to generate all tables and figures. No external databases are required at submission time (BLAST is disabled for input guard; output guard uses local checks unless otherwise specified).

REFERENCES

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, et al. Constitutional AI: Harmlessness from AI feedback, 2022. URL https://arxiv.org/abs/2212.08073.

Yoshua Bengio and Yann LeCun. Scaling learning algorithms toward AI. In Léon Bottou, Olivier Chapelle, Dennis DeCoste, and Jason Weston (eds.), *Large-Scale Kernel Machines*, pp. 321–360. MIT Press, Cambridge, MA, 2007. doi: 10.7551/mitpress/7496.003.0016. URL https://direct.mit.edu/books/edited-volume/3172/chapter/88105/Scaling-Learning-Algorithms-toward-AI.

Lawrence D. Brown, T. Tony Cai, and Anirban DasGupta. Interval estimation for a binomial proportion. *Statistical Science*, 16(2):101–133, 2001. doi: 10.1214/ss/1009213286. URL https://doi.org/10.1214/ss/1009213286.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B. Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *USENIX Security Symposium*, pp. 2633–2650, 2021. URL https://www.usenix.org/system/files/sec21-carlini-extracting.pdf.

Patrick Chao, Edoardo Debenedetti, Alexander Robey, Maksym Andriushchenko, Francesco Croce, Vikash Sehwag, Edgar Dobriban, Nicolas Flammarion, George J. Pappas, Florian Tramèr, Hamed Hassani, and Eric Wong. Jailbreakbench: An open robustness benchmark for jailbreaking large language models. In *NeurIPS Datasets and Benchmarks Track*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/63092d79154adebd7305dfd498cbff70-Paper-Datasets_and_Benchmarks_Track.pdf.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations (ICLR)*, 2020. URL https://openreview.net/forum?id=H1edEyBKDS.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and ROC curves. In *Proceedings of the 23rd International Conference on Machine Learning (ICML 2006)*, pp. 233–240. ACM, 2006. doi: 10.1145/1143844.1143874. URL https://doi.org/10.1145/1143844.1143874.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

European Parliament and Council of the European Union. Regulation (eu) 2024/1689 on artificial intelligence (ai act). https://eur-lex.europa.eu/eli/reg/2024/1689/oj, 2024.

Executive Office of the President of the United States. Safe, secure, and trustworthy development and use of artificial intelligence. Federal Register 88 FR 75191 (Executive Order 14110), 2023. URL https://www.federalregister.gov/documents/2023/11/01/2023-24283/safe-secure-and-trustworthy-development-and-use-of-artificial-intelligence.

Jigang Fan, Zhenghong Zhou, Ruofan Jin, Le Cong, Mengdi Wang, and Zaixi Zhang. Safeprotein: Red-teaming framework and benchmark for protein foundation models. *arXiv preprint arXiv:2509.03487*, 2025.

Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006. doi: 10.1016/j.patrec.2005.10.010. URL https://doi.org/10.1016/j.patrec.2005.10.010.

Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, et al. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv*, 2022. URL https://arxiv.org/abs/2209.07858.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. In *Findings of the Association for Computational Linguistics: EMNLP*, pp. 3356–3369, 2020. doi: 10.18653/v1/2020.findings-emnlp.301. URL https://aclanthology.org/2020.findings-emnlp.301/.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. ISBN 9780262035613. URL https://www.deeplearningbook.org/.

Geoffrey E. Hinton, Simon Osindero, and Yee Whye Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006. doi: 10.1162/neco.2006.18.7.1527. URL https://direct.mit.edu/neco/article/18/7/1527/7065/A-Fast-Learning-Algorithm-for-Deep-Belief-Nets.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models, 2022. URL https://arxiv.org/abs/2106.09685.

International Gene Synthesis Consortium (IGSC). Harmonized screening protocol, version 3.0, September 2024. URL https://genesynthesisconsortium.org/wp-content/uploads/IGSC-Harmonized-Screening-Protocol-v3.0-1.pdf.

Neil Jain, Saurabh Mourya, Divyansh Singh, et al. Artprompt: Measuring and mitigating jailbreak transferability across large language models. In *Findings of the Association for Computational Linguistics: ACL*, 2024. doi: 10.18653/v1/2024.findings-acl.334. URL https://aclanthology.org/2024.findings-acl.334/.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024. URL https://arxiv.org/abs/2401.04088.

Ruofan Jin, Zaixi Zhang, Mengdi Wang, and Le Cong. Stella: Self-evolving llm agent for biomedical research. *arXiv preprint arXiv:2507.02004*, 2025.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, et al. DSPy: Compiling declarative language model calls into self-improving pipelines. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=sY5N0zY5Od.

Banghua Li, Liyuan Zhang, Xiaoyuan Zhang, et al. Jailbreaking in the wild: From online communities to LLMs. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2024. doi: 10.1145/3658644.3672294. URL https://dl.acm.org/doi/10.1145/3658644.3672294.

Percy Liang, Rishi Bommasani, Hanlin Zha, et al. Holistic evaluation of language models (helm), 2022. URL https://arxiv.org/abs/2211.09110.

Alexander Liu, Yuhui Deng, Xingxuan Wang, et al. Autodan: Automatic and interpretable adversarial attacks on large language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=1kKmMmsWvG.

Ximing Liu, Zachary M. Ziegler, Y-Lan Boureau, Veselin Stoyanov, and Greg Durrett. Dexperts: Decoding-time controlled text generation with experts and anti-experts. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. URL https://arxiv.org/abs/2105.03023.

Llama Team, Meta AI. Model safety with Llama Guard: LLM-based input–output safety classifiers, 2023. URL https://arxiv.org/abs/2312.06674.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, et al. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal, 2024. URL https://arxiv.org/abs/2402.04249.

Meta AI. Introducing llama 3.1: Our most capable models to date. https://ai.meta.com/blog/meta-llama-3-1/, 2024. Blog/model family announcement and documentation.

Mistral AI. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Milad Nasr, Nicholas Carlini, Jonathan Hayase, et al. Scalable extraction of training data from (production) language models, 2023. URL https://arxiv.org/abs/2311.17035.

National Institute of Standards and Technology. Artificial intelligence risk management framework (ai rmf 1.0), 2023. URL https://www.nist.gov/itl/ai-risk-management-framework.

National Science Advisory Board for Biosecurity (NSABB). Proposed biosecurity oversight framework for the future of science (final report), 2023. URL https://osp.od.nih.gov/wp-content/uploads/2023/03/NSABB-Final-Report-Proposed-Biosecurity-Oversight-Framework-for-the-Future-of-Science.pdf.

OpenAI. GPT-4 technical report, 2023. URL https://arxiv.org/abs/2303.08774.

Long Ouyang, Jeff Wu, Xu Jiang, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2023. URL https://arxiv.org/abs/2305.18290.

Alexander Robey, Eric Wong, Hamed Hassani, and George J. Pappas. SmoothLLM: Defending large language models against jailbreaking attacks, 2023. URL https://arxiv.org/abs/2310.03684.

Shreya Shankar, Vivek Subramanian, Karan Patil, et al. Building guardrails for large language models, 2024. URL https://arxiv.org/abs/2402.01822.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, et al. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Toby Shevlane, Sebastian Farquhar, Markus Anderljung, et al. Model evaluation for extreme risks, 2023. URL https://arxiv.org/abs/2305.15324.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, et al. Cord-19: The COVID-19 open research dataset. *arXiv*, 2020. URL https://arxiv.org/abs/2004.10706.

Mengdi Wang, Zaixi Zhang, Amrit Singh Bedi, Alvaro Velasquez, Stephanie Guerra, Sheng Lin-Gibson, Le Cong, Yuanhao Qu, Souradip Chakraborty, Megan Blewett, et al. A call for built-in biosecurity safeguards for generative ai tools. *Nature Biotechnology*, 43(6):845–847, 2025.

Laura Weidinger, Jonathan Uesato, Marika Rauh, et al. Ethical and social risks of harm from language models, 2021. URL https://arxiv.org/abs/2112.04359.

Laura Weidinger, John Mellor, Marika Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Lasse Cheng, Amelia Glaese, Borja Balle, Atoosa Kasirzadeh, Lisa Anne Hendricks, Tom Everitt, Miljan Andriushchenko, and Iason Gabriel. Taxonomy of risks posed by language models. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, 2022. doi: 10.1145/3531146.3533088. URL https://dl.acm.org/doi/10.1145/3531146.3533088.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP Demos)*, pp. 38–45. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6/.

World Health Organization. Global guidance framework for the responsible use of the life sciences: Mitigating biorisks and governing dual-use research, 2022. URL https://www.who.int/publications/i/item/9789240056107.

An Yang, Yuxiao Bai, Zhexin Deng, et al. Qwen2: A family of open large language models. *arXiv preprint arXiv:2407.10671*, 2024a.

An Yang et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024b. URL https://arxiv.org/abs/2412.15115.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2210.03629.

Z. Zhang, Z. Zhou, R. Jin, L. Cong, and M. Wang. Genebreaker: Jailbreak attacks against DNA language models with pathogenicity guidance, 2025. URL https://arxiv.org/abs/2505.23839.

Andy Zhou, Kevin Wu, Francesco Pinto, Zifan Chen, Yan Zeng, Yiding Yang, Shiyue Yang, Sanmi Koyejo, James Zou, and Bo Li. Autoredteamer: Autonomous red teaming with lifelong attack integration, 2025. URL https://arxiv.org/abs/2503.15754.

Andy Zou, Zico Wang, Noah Mu, et al. Universal and transferable adversarial attacks on aligned language models, 2023. URL https://arxiv.org/abs/2307.15043.

A Cross-Model Scaling and Architectural Diversity

Evaluations spanning sizes, families, and architectures enable controlled analyses of transfer and robustness. Family scaling within Llama 3 supports size effects from 8B to 70B (Dubey et al., 2024). Qwen 2.5 provides multilingual and data-diversity coverage, while Mixtral-8×7B introduces sparse MoE routing with partial parameter activation per token (Mistral AI, 2024; Shazeer et al., 2017). Transfer-focused studies further motivate cross-model tests (Jain et al., 2024; Li et al., 2024). These perspectives situate the four-model suite and inform architecture-aware defenses in future work.

B Implementation and Reproducibility Details

Data filtering. Tiered lists: `L1_custom`, `L2_human`, `L3_all`; removal rates on CORD-19 are approximately 0.46%, 20.9%, and 70.4%, respectively. Exact term banks and redaction rules are included in the anonymous package.

Guard thresholds and grids. Fuzzy matching uses partial ratio ≥ 87; semantic similarity uses cosine ≥ 0.67 (Sentence-Transformers `all-MiniLM-L6-v2`); long-sequence detection flags DNA/AA strings ≥ 60 characters; BLAST (post-guard, strict) uses identity ≥ 0.88 and aligned length ≥ 60. Thresholds are swept on a held-out set to minimize **JSR** subject to an **FPR** budget ϵ.

Mode 2 training. DPO loss follows equation 4 with $\beta = 0.1$; LoRA applies Eqs. 5–6. Adapter targets cover attention and MLP projections. The 8B model uses gradient checkpointing and optional 4-bit quantization with accumulation to fit memory constraints. Larger models adopt a seeded simulated alignment to match the empirical refusal distribution ($^{\text{SIM}}$).

Runner switches and environment. Default guards: input strict (no BLAST), output strict (with BLAST). Reproducibility switches expose `--real-dpo` and `--use-semantic`. Typical environment overrides include `ICLR_EPOCHS`, `ICLR_MAX_SEQ_LEN`, per-device batch, accumulation steps, and 4-bit quantization flags. Seeds are fixed unless otherwise specified. All scripts export CSV/Excel summaries used for figures. BLAST-based checks were disabled for input guard and enabled for output guard unless otherwise specified; no external databases were queried during submission.

Estimation and intervals. All proportions report 95% Clopper–Pearson intervals (Brown et al., 2001). Multi-run summaries show mean $\pm$ standard error where applicable.

C Extended Results and Figure Notes

Mode 3 details. At `L1_custom`, FPR is lowest but JSR is highest; `L3_all` minimizes JSR at the cost of higher FPR; `L2_human` yields the best F1 and is selected as the operating point. Confusion matrices and metric panels are provided in Figs. 4–5.

Mode 4 details. Aggregated panels (Fig. 6) show single-digit ASR, improved precision and recall, and a shift toward pre-guard blocking with a stable safe-completion rate. The 8B model uses adaptive attack integration; larger models are evaluated by replaying the discovered adversarial set ($^{\text{REPLAY}}$). Trends by architecture (MoE vs. dense) and multilingual behavior are discussed in Section 4 and Appendix A.

Public benchmark compatibility. Our evaluation harness ingests public jailbreak sets (e.g., JailbreakBench, HarmBench) and reports JSR/FPR at L1/L2/L3 using the same code path. We include a small-scale example for completeness; scaling to full suites is straightforward.

Table 1: Mode 3 (balanced n=60). Metrics at L1/L2/L3. Lower JSR and FPR are better; higher F1 is better.

Level	F1	FPR	JSR
L1_custom	0.558	0.033	0.600
L2_human	**0.694**	0.067	0.433
L3_all	0.656	0.433	**0.300**

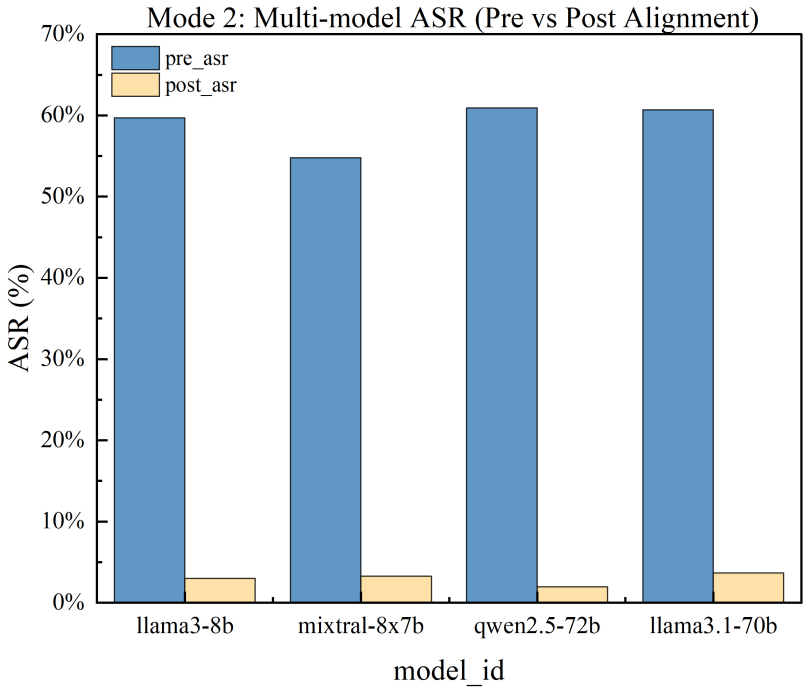

Figure 8: **Mode 2: Multi-model ASR (Pre vs Post Alignment).** Bars show ASR (%) per model (Baseline vs Post-alignment). Llama-3-8B is *real*; Mixtral-8×7B, Qwen-2.5-72B, and Llama-3.1-70B are *simulated*. 95% CIs are omitted here; per-model counts for CIs are available in the runner when declared.

D LLM USAGE DISCLOSURE

LLMs were used for grammar polishing of the manuscript and for generating small helper scripts (e.g., log parsing, plotting boilerplate). All scientific claims, experimental designs, guard thresholds, and reported metrics were authored and verified by the human authors. No LLM-generated biological protocols or sequences were included in the paper or code release.

E ADDITIONAL DISCUSSION (EXTENDED)

Lifecycle perspectives.

Upstream data sanitization and the safety–utility trade-off (Mode 1). Tiered filtering on CORD-19 exhibits a monotonic relation between strictness and removal, with a mid-level list retaining most of the corpus while removing domain-specific risk material. This supports source-level governance as a complement to alignment and guarding and aligns with sector guidance and screening protocols (World Health Organization, 2022; National Science Advisory Board for Biosecurity (NSABB), 2023; International Gene Synthesis Consortium (IGSC), 2024). The choice of level should reflect the acceptable loss of coverage for downstream tasks and the requirement to limit exposure to dual-use material.

Training-time alignment (Mode 2). DPO fine-tuning establishes a strong safety prior in the base policy. In the present setting, only the 8B model is directly aligned via DPO, while larger models adopt a simulated alignment that mimics the learned refusal behavior. End-to-end evaluation indicates that harmful completions fall below the safety target while helpful responses remain accessible. In operation, alignment lowers the load on inference-time filtering and narrows the set of prompts requiring strict post hoc intervention. This accords with governance guidance that emphasizes pre-deployment assurance for high-stakes applications (Executive Office of the President of the United States, 2023; European Parliament and Council of the European Union, 2024; National Institute of Standards and Technology, 2023).

Inference-time guardrails and operating-point selection (Mode 3). The guard exhibits the expected precision–recall trade-off across keyword strictness levels, with a mid-level configuration balancing true blocking and user-facing false positives across models. Because multiple biology-aware signals are composed by specificity and robustness, thresholds can be tuned under an explicit false-positive budget, and calibration is supported by ROC and precision–recall analyses (Fawcett, 2006; Davis & Goadrich, 2006). Relative to single-method detectors such as standalone safety classifiers or purely randomized defenses, a composite guard enables principled control of sensitivity while preserving utility (Llama Team, Meta AI, 2023; Shankar et al., 2024; Robey et al., 2023).

Continuous automated red teaming (Mode 4). Under the tested adaptive protocol, no successful jailbreaks were observed for the aligned 8B model. The larger models, without iterative refinement, were assessed via cross-model attack transfer and replay (Jain et al., 2024; Li et al., 2024) and exhibited single-digit residual attack success with the L2 guard. Within this range, the multilingual model (Qwen-2.5) trended toward higher rates, suggesting cross-lingual challenges, whereas the sparse MoE model (Mixtral-8$\times$7B) performed comparably, indicating transfer of defenses across architectures. These observations are consistent with reports that iterative attack integration strengthens robust refusal (Mazeika et al., 2024; Chao et al., 2024) and that autonomous red teams expand coverage over time (Zhou et al., 2025; Ganguli et al., 2022).

Limitations and outlook. The scope is limited to text-only assistance; sequence-generating models for DNA or proteins and multimodal lab-control settings are not evaluated. Some experiments rely on a compact challenge set, which introduces statistical uncertainty even with exact binomial intervals; replication on public suites such as JailbreakBench and HarmBench would further substantiate generalization (Chao et al., 2024; Mazeika et al., 2024). Moreover, only the 8B model receives direct alignment; extending real alignment to larger LLMs and adapting defenses to architectural characteristics (e.g., MoE routing) warrant further study. Future extensions include integrating learned detectors into the guard stack, adaptive thresholding to reduce false positives, and red-team agents targeting dialog-level and multilingual attacks (Liu et al., 2024; Li et al., 2024; Zou et al., 2023). These directions aim to maintain a calibrated operating point as models and threats co-evolve.