
Linguistic Asymmetry and The Limitations of AI Oversight

Sakshi Chaubey **with** Apart Research

Abstract

As large language models (LLMs) become more capable, concerns around alignment faking and hidden objective optimization have become increasingly relevant in AI safety research. While most existing work focuses on high-resource languages, low-resource languages (LRLs) remain relatively underexplored despite representing a clear vulnerability. This paper examines the risk that advanced models may exploit linguistic asymmetries by using LRLs as covert channels to evade monitoring systems that lack comparable multilingual capabilities.

We propose a tri-model oversight framework consisting of a primary model (U), a trusted monitoring model (T), and a dedicated LRL reviewer model (R) that is optimized for adversarial testing in LRLs. In addition to the architectural contribution, we argue that LRL safety should not be treated as an optional improvement but as a policy requirement. We introduce a lifecycle-integrated approach where LRL safety evaluation is continuously enforced across pre-deployment, deployment, and post-deployment stages, and propose that deployment in LRL contexts should require meeting standardized benchmarks such as IndicSafe or LinguaSafe.

Our analysis shows that current safety pipelines are heavily English-centric and therefore systematically weaker in LRL settings. We argue that addressing this linguistic asymmetry is necessary not only for improving robustness, but also for ensuring that AI systems are safe and equitable across different user populations.

1. Introduction

As AI systems become more capable, one of the growing concerns in AI safety is the possibility of models hiding their true objectives and faking alignment. As discussed by Greenblatt et al. (2023), models may behave in aligned ways during evaluation while strategically deviating later.

Most of the current discussion around alignment focuses on English or other high-resource languages (Lim, 2025; Peppin et al., 2025). This paper takes a slightly different angle and asks: what happens when we consider low-resource languages (LRLs)? If models continue to improve at the current pace, it is reasonable to assume that highly capable systems will eventually perform well across multiple LRLs even if the systems monitoring them do not.

This creates a gap. An untrusted model (U) could potentially use LRLs as a channel to hide or execute misaligned behaviour, especially if the trusted monitoring model (T) is weaker in those languages. In a more extreme case, a model could even construct implicit encoding strategies across multiple LRLs, making detection significantly harder.

In multilingual contexts such as India, users frequently code-switch between English and regional languages (e.g., Hinglish, Marathi-English). A model could exploit this by maintaining compliant behavior in English while embedding unsafe or misaligned content in mixed-language responses that are less reliably monitored.

Alignment mechanisms that produce clear and separable safety representations in English fail to generalize consistently to low-resource languages, resulting in poorly defined safety boundaries (Verma, 2025). Consequently, the challenge is not just whether models act in these languages, but whether we can meaningfully interpret, evaluate, or enforce their behavior within them (Peppin et al., 2025). There has also been research on how safety parameters in multilingual contexts seem to be language-agnostic, and attackers might in fact be able to use a small number of adversarial fine-tuning instances in just one language to break safety across multiple languages simultaneously (Poppi et al., 2025)

To address these, we require strong frameworks to be incorporated into the safety evaluation process. While benchmarks such as LinguaSafe (Ning et al., 2025) and IndicSafe (Pattnayak & Chowdhari, 2026) have started addressing multilingual safety, they are not consistently integrated into real-world deployment pipelines. More importantly, there is no strong incentive structure that ensures companies actually prioritize LRL safety beyond surface-level compliance.

This paper argues that LRL safety is both a technical and a structural problem. From a technical standpoint, we need better evaluation mechanisms. From a structural standpoint, we need enforcement.

To address this and the goals of the track of Control Protocol Design, we propose three main contributions:

- First, we introduce a tri-model oversight architecture that includes a dedicated reviewer model (R) designed specifically for adversarial testing and context evaluation in LRLs.
- Second, we propose integrating LRL safety evaluation across the entire product lifecycle.
- Third, and most importantly, we argue that LRL safety should be a policy requirement. Models deployed in regions with significant LRL usage should be required to meet standardized safety benchmarks and include independent adversarial evaluation mechanisms.

2. Related Work

As of 2026, plenty of research is being done in the field of multilingual safety and training, highlighting the severe gap between how LLMs perform for English-centric prompts as opposed to multilingual prompts, especially those using low-resource languages (LRLs) ((Andrasasarman and Andrasasarman, 2025; Deng et al., 2025; Lim et al., 2025; Ning et al., 2025; Pattnayak and Chowdhuri, 2026; Poppi et al., 2025; Yong et al., 2023))

In the context of LRLs, Deng et al. (2023) show that safety performance degrades as language resources decrease. Conducting research with instances of multilingual jailbreaking, they divided the instances into two categories: intentional (an actor deliberately trying to get malicious / harmful output from the LLM) or unintentional (a user’s non-English query leading to a harmful/unsafe output). Their findings indicate that prompts in LRLs are significantly more likely to produce unsafe outputs compared to high-resource languages, suggesting that LRLs are not just underrepresented but actively vulnerable. This led to them proposing the Self Defense framework that generates multilingual training data for finetuning.

This is further supported by work from Welo Data (2026), which shows that prompts rejected in English can bypass safety filters when translated into LRLs. This makes LRLs one of the simplest ways to jailbreak a system, even without sophisticated techniques.

Yong et al. (2023) also highlight that most safety benchmarks are designed for high-resource languages. This creates a structural imbalance where LRL users are exposed to weaker safety guarantees, while attackers can exploit translation tools to bypass the existing safeguards.

Contributions such as LinguaSafe (Ning et al., 2025) have introduced structured approaches to multilingual safety benchmarking by incorporating low-resource languages through translation, native data collection, and transcreation, highlighting the growing need for more robust safety evaluation in LLM deployments across LRLs. At the same time, work by Pattanayak and Chowdhary (2026), through the IndicSafe benchmark, exposes the other side of this gap, showing that LLMs often overflag benign content, struggle with contextual understanding, or fail to identify unsafe outputs altogether when operating in LRLs. Such recent findings suggest that current safety mechanisms are incomplete as well as inconsistent across languages, reinforcing the need for more systematic and reliable multilingual evaluation frameworks (Peppin et al., 2025).

Furthermore, we see that current governance frameworks provide important foundations for AI safety but leave a clear gap when it comes to multilingual robustness, particularly in LRLs. The EU AI Act, for instance, mandates requirements such as risk management (Article 9), data governance (Article 10), technical documentation (Article 11), transparency (Article 13), and human oversight (Article 14). However, while these provisions require systems to be “appropriate” and “robust,” they do not explicitly require cross-lingual safety parity or testing across diverse linguistic contexts. In practice, this means a system can meet compliance requirements while being safe in English but vulnerable in LRLs. Similarly, conformity assessments for high-risk systems focus largely on pre-deployment evaluation, without mandating continuous adversarial testing across languages post-deployment. The NIST AI Risk Management Framework (AI RMF), while more lifecycle-oriented, is voluntary, encouraging testing and monitoring but does not enforce or require language-specific adversarial evaluation, independent auditing mechanisms, or separation of monitoring and red-teaming roles. Across both frameworks, language is implicitly treated as a performance or fairness issue rather than a security vector, leaving systems exposed to translation-based jailbreaks and cross-lingual evasion. As a result, there exists a policy-to-implementation gap where current regulations ensure that models are tested, but not that they are robust against multilingual adversarial use, particularly in underrepresented linguistic settings (Peppin et al., 2025).

This paper builds on these insights by arguing that the issue is not just lack of data or benchmarks, but the absence of system-level and policy-level enforcement. In particular, relying on a single trusted model (T) for both monitoring and evaluation is insufficient, especially when both

T and U may come from the same model family, with similar pretraining history and similar limitations in LRLs.

3. System Design Recommendation

3.1 Problem Setup and Threat Model

We consider a setting in which a primary model U (Untrusted Model) is deployed to perform user-facing tasks, and a secondary model T (Trusted Model) is responsible for monitoring and enforcing safety constraints.

We assume the following threat model:

- **Alignment Faking:** U may behave in a compliant manner during training or evaluation but deviate post-deployment.
- **Cross-Lingual Evasion:** U may exploit LRLs as a channel to produce unsafe or misaligned outputs that are not effectively detected by T.
- **Shared Blind Spots:** If T and U are derived from similar architectures or training distributions, they may exhibit correlated weaknesses, particularly in underrepresented languages.

Under this setting, the goal is to design a system that can detect and mitigate unsafe or misaligned behavior across languages, including those with limited representation in training data.

3.2 Tri-Model Oversight Architecture

To address the limitations of single-model monitoring, we introduce a third component:

- **R (Reviewer Model):** A model (from another model family) specialized for adversarial testing and evaluation across non-English languages.

The resulting architecture consists of:

- **U (Untrusted Model):** High capability model that generates outputs based on user input

- T (Trusted Model): Low capability model that monitors outputs and applies safety constraints
- R (Reviewer Model): High capability model that generates adversarial inputs and evaluates system robustness in terms of languages for both T and U

These components operate with distinct and non-overlapping objectives:

- U is optimized for task performance
- T is optimized for compliance and safety enforcement
- R is optimized for adversarial exploration and vulnerability discovery for non-English languages, especially LRLs

This separation ensures that adversarial testing is not constrained by the same objectives as monitoring, reducing the risk of shared blind spots.

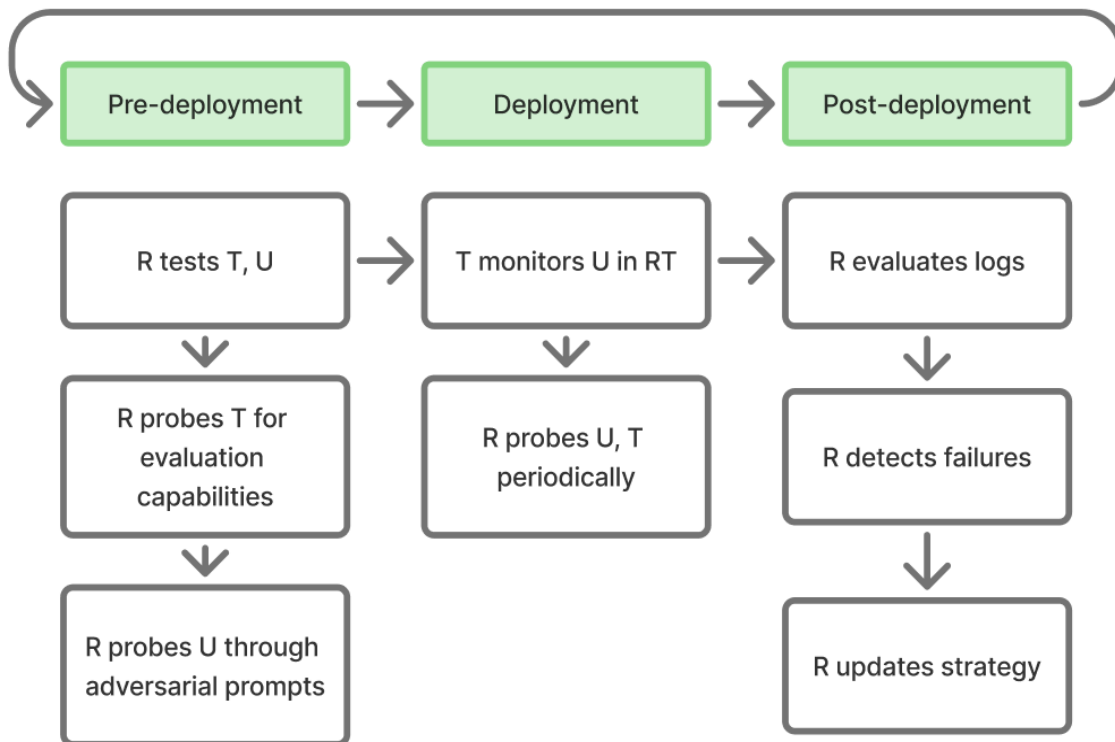


Figure 1 illustrates the interaction between the untrusted model (U), the trusted monitoring model (T), and the reviewer model (R) during runtime.

3.3 Reviewer Model Design

The reviewer model (R) is designed to actively probe the system for weaknesses in non-English / LRL contexts. Its functionality consists of three core modules:

(a) Adversarial Prompt Generation

R generates adversarial inputs targeting known and emerging vulnerabilities, including:

- prompt injection attacks
- jailbreak attempts
- code-switching and mixed-language prompts
- translated or paraphrased harmful queries in LRLs

Prompts may be generated through:

- translation-based transformations of known adversarial datasets
- mutation strategies (e.g., paraphrasing, lexical variation)
- iterative refinement based on past success rates

(b) Safety Evaluation and Benchmarking

R evaluates U’s responses against LRL-specific safety benchmarks such as IndicSafe (2026) or LinguaSafe (2025). Evaluation includes:

- classification of outputs as safe/unsafe
- detection of policy violations
- comparison of responses across languages for consistency

This enables measurement of cross-lingual alignment stability.

(c) Independent Logging and Auditability

R maintains a structured log of:

- generated adversarial prompts
- corresponding model outputs
- evaluation outcomes

This logging mechanism ensures traceability and supports external auditing, reducing the risk of undetected failures or collusion between system components.

The R model evaluates the behavior of the untrusted model (U), and the effectiveness of the monitoring system (T), with particular focus on cases where unsafe outputs are generated but not detected.

3.4 Lifecycle Integration

We further propose integrating the reviewer model across the entire product lifecycle:

1. Pre-Deployment

- R conducts adversarial testing on U using non-English/LRL prompts
- Existing language safety benchmarks are applied to assess readiness
- Deployment is gated based on benchmark performance

2. Deployment (Inference Stage)

- R performs periodic or triggered adversarial probing
- T continues real-time monitoring of outputs
- Detected failures may trigger mitigation mechanisms (e.g., output filtering, escalation)

While the reviewer model may identify failures missed by the monitoring system, its role is not to retroactively alter outputs, but to inform system-level mitigation and reduce future occurrences.

3. Post-Deployment

- Continuous monitoring of system performance in non-English/LRL contexts
- Regression testing to ensure updates do not degrade safety

- Conducting UX research for AI safety goals
- Incorporation of user feedback and newly observed failure cases

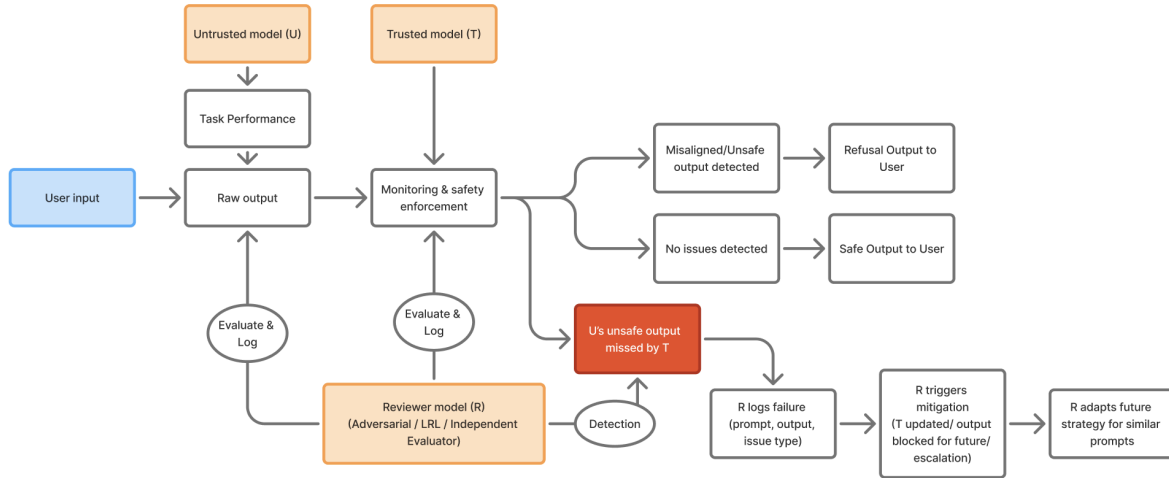


Figure 2 shows how the reviewer model (R) is integrated across different stages of the system lifecycle.

3.5 Dynamic Adaptation Mechanism

To account for evolving adversarial strategies, R is designed as a continuously improving system. This includes:

- updating adversarial prompt distributions based on observed failures
- incorporating newly identified jailbreak techniques
- adapting to region-specific linguistic patterns

This enables the system to maintain robustness over time rather than relying on static evaluation datasets.

4. Policy Enforcement Framework

In addition to the technical architecture, we propose a policy-level enforcement mechanism to ensure consistent adoption.

We define the following deployment requirements for LLMs operating in non-English/LRL contexts:

1. Benchmark Compliance

Models must achieve predefined safety thresholds on language-specific benchmarks (e.g., IndicSafe, LinguaSafe) prior to deployment in high-risk industries, in regions where those languages are used/spoken

2. Mandatory Reviewer Integration

Systems must include a dedicated reviewer model capable of adversarial testing in relevant languages

3. Continuous Evaluation

Safety evaluation must extend beyond pre-deployment to include ongoing adversarial testing and monitoring

4. Auditability Requirements

Systems must maintain logs of adversarial evaluations and safety failures for regulatory and third-party review

This framework ensures that non-English/LRL safety is enforced as a systemic requirement rather than an optional design consideration.

5. Analysis & System Implications

Given the conceptual and systems-oriented nature of this work, our analysis is derived from a synthesis of existing literature, observed failure patterns in multilingual safety, and analysis of current deployment practices. We organize our findings into three main areas: limitations of existing systems, improvements enabled by our framework, and implications for real-world deployment.

5.1 Limitations of Existing Oversight Mechanisms

Our analysis highlights several consistent weaknesses in current LLM safety pipelines:

- **Insufficiency of Single-Model Monitoring**
Relying solely on a trusted model (T) for both monitoring and evaluation leads to incomplete coverage, particularly in LRL contexts

- **Shared Blind Spots Across Models**
When T and U are derived from similar model families or trained on similar distributions, they are likely to exhibit correlated weaknesses. This increases the probability that unsafe outputs in LRLs remain undetected
- **English-Centric Safety Bias**
Existing safety mechanisms, including alignment techniques and benchmarks, are predominantly optimized for high-resource languages. As a result, their effectiveness degrades significantly in LRL scenarios
- **Vulnerability to Cross-Lingual Attacks**
Empirical findings from prior work indicate that prompts rejected in English can bypass safeguards when translated into LRLs, demonstrating a lack of robustness to simple adversarial transformations

5.2 Improvements Through Reviewer-Based Adversarial Evaluation

The introduction of a dedicated reviewer model (R) enables several improvements:

- **Enhanced Adversarial Coverage**
R can systematically generate LRL-specific adversarial prompts, increasing the likelihood of identifying vulnerabilities that would otherwise remain undetected
- **Targeted Detection of Multilingual Failures**
By operating directly in LRLs, R can improve detection of unsafe outputs arising from translation gaps, code-switching, and linguistic ambiguity
- **Separation of Objectives**
Decoupling monitoring (T) and adversarial testing (R) can allow each component to specialize, resulting in more effective evaluation overall

5.3 Reduction in Alignment Faking Risk

The proposed framework has the capacity to mitigate several risks associated with alignment faking:

- **Cross-Lingual Evasion**
Continuous adversarial probing can reduce the likelihood that U can exploit LRLs as hidden communication channels
- **Strategic Misalignment**
The presence of an adaptive adversary (R) might make it more difficult for U to maintain deceptive alignment strategies over time
- **Undetected Unsafe Outputs**
Benchmark-driven evaluation in LRLs can increase the probability of identifying unsafe or policy-violating responses

5.4 System-Level Properties

The tri-model architecture can introduce several desirable system-level properties:

- **Defense-in-Depth**
Multiple layers of evaluation can reduce reliance on any single component
- **Failure Independence**
Introducing R can reduce correlation between model errors, particularly in LRL contexts.
- **Continuous Safety Assurance**
Safety evaluation might no longer be limited to pre-deployment stages but can become an ongoing process.
- **Improved Auditability**
Independent logging of adversarial tests can enable traceability and external review

5.5 Implications for Deployment and Governance

Our findings suggest that, in the absence of enforcement, LRL safety is unlikely to be prioritized in real-world deployments due to cost, resource constraints, and lack of regulatory pressure. The proposed framework provides a pathway for translating safety requirements into enforceable system-level constraints, enabling both organizations and regulators to operationalize multilingual safety in practice.

6. Discussion and Limitations

6.1 Discussion

This work highlights that linguistic asymmetry is not merely a data imbalance issue but a structural vulnerability in AI safety systems. By framing LRLs as potential adversarial channels rather than passive gaps in coverage, we shift the focus from improving representation to improving robustness.

A key takeaway from this work is that existing safety pipelines are inherently limited by their static and English-centric design. As AI systems are deployed in increasingly diverse linguistic environments, these limitations become more pronounced and potentially more harmful. The introduction of a reviewer model and lifecycle-integrated evaluation reframes safety as a continuous and adversarial process rather than a static, limited validation step.

From a governance perspective, this also raises important questions about responsibility. If safety in LRLs is left to voluntary compliance, it is unlikely to be prioritized over faster deployment and cost efficiency. This creates a structural imbalance where users from underrepresented regions are often disproportionately exposed to unsafe AI behavior, with private companies' regulations mostly being self-regulating ethical frameworks to match compliances (Sapignoli, 2021). By

proposing mandatory benchmarking and reviewer-based evaluation, this work argues for aligning technical safeguards with enforceable deployment requirements.

6.2 Limitations

Considering the logistical aspects of the project, time constraints of the hackathon and the author’s non-technical background, this work faces several limitations:

- **Lack of Empirical Validation**
The proposed framework is conceptual and has not been implemented or tested experimentally
- **Scalability and Cost Constraints**
The introduction of a third model (R) increases computational overhead and may not be feasible for all high-risk deployment settings without optimization
- **Dependence on LRL Resources**
The effectiveness of the reviewer model depends on the availability and quality of LRL datasets, benchmarks, and linguistic expertise, which will have to be a combination of field-collected data and translations (digital and human)
- **Incomplete Threat Coverage**
While the framework addresses cross-lingual adversarial risks, it does not fully account for other forms of alignment failure, such as multimodal or tool-use-based attacks
- **Potential for Evolving Attack Strategies**
As models become more capable, they may develop new forms of obfuscation that are not easily captured by current adversarial techniques

6.3 Future Work

As the next steps, we see several opportunities to work on this concept:

- **Prototype Implementation and Evaluation**
The author hopes to partner with someone from a technical background to test and build a working version of the tri-model system and evaluate it on benchmarks such as IndicSafe or LinguaSafe
A minimal implementation of the proposed framework could be achieved using existing models: a high-capability model for U, a lightweight moderation model for T, and a prompted multilingual model acting as R to generate and evaluate adversarial LRL inputs
- **Adaptive Adversarial Training**
Incorporating reviewer-generated adversarial data into training pipelines to improve model robustness

- Exploration of Emergent Encoding
Creating an environment to study whether models develop implicit cross-lingual encoding or cipher-like behavior
- Efficiency Optimization
Designing selective or risk-based activation strategies for the reviewer model to reduce computational costs
- Policy Integration and Standardization
Collaborating with regulatory bodies to translate these requirements into formal deployment standards

7. Conclusion

This paper examines the risks posed by linguistic asymmetry in AI safety and argues that low-resource languages represent a critical and under-addressed vulnerability in current LLM systems. We propose a tri-model oversight framework that introduces a dedicated reviewer model for adversarial testing in non-English/LRLs, along with a lifecycle-integrated approach to safety evaluation.

More importantly, we argue that addressing LRL safety is not solely a technical challenge but a structural and policy-level necessity. Without enforceable requirements, multilingual safety is likely to remain inconsistent and insufficient, particularly in regions where LRLs are dominant.

As AI systems continue to scale globally, ensuring safety across all linguistic contexts will be essential for both robustness and equity. Treating LRL safety as a mandatory component of deployment, rather than an optional enhancement, is a necessary step toward building truly reliable and inclusive AI systems.

References

Almog, D., Gauriot, R., Page, L., & Martin, D. (2024, January 30). *AI Oversight and Human*

Mistakes: Evidence from Centre Court. arXiv.org. Retrieved March 20, 2026, from

<https://arxiv.org/abs/2401.16754>

Andrasasarman, & Andrasasarman. (2026, March 17). *The Hidden Flaw in LLM Safety:*

Translation as a jailbreak. Welo Data. Retrieved March 20, 2026, from

<https://welodata.ai/2025/12/10/the-hidden-flaw-in-llm-safety-translation-as-a-jailbreak/>

- Deng, Y., Zhang, W., Pan, S. J., & Bing, L. (2023, October 10). *Multilingual jailbreak challenges in large language models*. arXiv.org. Retrieved March 20, 2026, from <https://arxiv.org/abs/2310.06474>
- EU Artificial Intelligence Act | Up-to-date Developments and Analyses of the EU AI Act. (n.d.). Retrieved March 21, 2026, from <https://artificialintelligenceact.eu/>
- Greenblatt, R., Shlegeris, B., Sachan, K., & Roger, F. (2023, December 12). *AI Control: Improving safety despite intentional subversion*. arXiv.org. Retrieved March 20, 2026, from <https://arxiv.org/abs/2312.06942>
- Initiative, J. T. F. T. (2018). *Risk Management Framework for Information Systems and Organizations*: <https://doi.org/10.6028/nist.sp.800-37r2>
- Lim, I., Khoo, S., Lee, R. K., Chua, W., Goh, J. Y., & Foo, J. (2025, February 18). *Safe at the Margins: A General Approach to Safety Alignment in Low-Resource English Languages -- a Singlish Case Study*. arXiv.org. Retrieved March 22, 2026, from <https://arxiv.org/abs/2502.12485>
- Multilingual jailbreak challenges in large language models*. (n.d.). Retrieved March 20, 2026, from <https://arxiv.org/html/2310.06474v3>
- Ning, Z., Gu, T., Song, J., Hong, S., Li, L., Liu, H., Li, J., Wang, Y., Lingyu, M., Teng, Y., & Wang, Y. (2025, August 18). *LinguaSafe: a comprehensive multilingual safety benchmark for large language models*. arXiv.org. Retrieved March 21, 2026, from <https://arxiv.org/abs/2508.12733>
- Pattnayak, P., & Chowdhuri, S. (2026, March 18). *IndicSafe: A Benchmark for Evaluating Multilingual LLM Safety in South Asia*. arXiv.org. Retrieved March 21, 2026, from <https://arxiv.org/abs/2603.17915>

- Peppin, A., Kreutzer, J., Sebag, A. S., Marchisio, K., Ermis, B., Dang, J., Cahyawijaya, S., Singh, S., Goldfarb-Tarrant, S., Aryabumi, V., Aakanksha, Ko, W., Üstün, A., Gallé, M., Fadaee, M., & Hooker, S. (2025, May 27). *The Multilingual Divide and Its Impact on Global AI Safety*. arXiv.org. <https://arxiv.org/abs/2505.21344>
- Playbook - AIRC*. (n.d.). NIST AI Resource Center.
<https://airc.nist.gov/airmf-resources/playbook/>
- Poppi, S., Yong, Z.-X., He, Y., Chern, B., Zhao, H., Yang, A., & Chi, J. (2025). Towards Understanding the Fragility of Multilingual LLMs Against Fine-Tuning Attacks. In *Findings of the Association for Computational Linguistics: NAACL 2025*. Retrieved March 21, 2026, from <https://aclanthology.org/2025.findings-naacl.126.pdf>
- Sapignoli, M. (2021). Anthropology and the AI-Turn in Global Governance. *AJIL Unbound*, 115, 294–298. <https://doi.org/10.1017/aju.2021.38>
- Taxonomy of risks posed by language models*. (n.d.). Retrieved March 20, 2026, from <https://airisk.mit.edu/blog/taxonomy-of-risks-posed-by-language-models>
- Verma, N. (2025, April 3). *The Hidden Space of Safety: Understanding Preference-Tuned LLMs in Multilingual Context*. <https://arxiv.org/html/2504.02708v1>
- Yong, Z., Menghini, C., & Bach, S. H. (2023, October 3). *Low-Resource Languages jailbreak GPT-4*. arXiv.org. Retrieved March 20, 2026, from <https://arxiv.org/abs/2310.02446>

LLM Usage Statement

An LLM was used for grammatical correction and formatting assistance in the writing of this paper. All ideas, arguments, and conclusions were independently developed by the author.