

Zen3-Guard: A Packaging of IBM Granite Guardian 8B

for the Zen Model Family

Technical Whitepaper v2025.06

Zach Kelling
Zen LM Research Team
research@zenlm.org
papers.zenlm.org

June 2025

Abstract

Zen3-Guard is a repackaging of IBM’s **Granite Guardian 8B** safety model, deployed as the content-safety layer for the Zen model family. It is *not* a from-scratch model: it is the openly released, Apache-2.0 licensed `ibm-granite/granite-guardian` 8B guardrail produced by IBM Research, which is itself a supervised fine-tune of the dense, decoder-only IBM Granite 3.x 8B Instruct base model (`GraniteForCausalLM`). Granite Guardian classifies a user prompt and/or assistant response against a configurable risk definition and emits a **Yes/No** verdict together with a probability of harm derived from the next-token logits. The risk taxonomy, aligned with the IBM AI Risk Atlas, spans harmful content (harm, social bias, profanity, sexual content, unethical behavior, violence, jailbreaking) as well as retrieval-augmented-generation faithfulness risks (context relevance, groundedness, answer relevance). IBM reports an aggregate AUC of **0.871** on harmful-content benchmarks and **0.854** on RAG-hallucination benchmarks for Granite Guardian 3.0 8B [4]. This document describes how the upstream model is integrated as a drop-in safety filter for Zen deployments; all model weights, training, and reported metrics are IBM’s and are cited as such. We add no benchmark claims of our own.

Contents

1	Introduction	2
2	Architecture	2
2.1	Upstream Model: Granite Guardian 8B	2
2.2	Risk Taxonomy and Output Schema	3
2.3	Language Coverage	3
2.4	Risk-Definition Prompting	3
3	Training	4
3.1	Provenance, Not Training	4
3.2	Calibration	4
4	Evaluation	4
4.1	Aggregate Results Reported by IBM	4
4.2	A Note on the Numbers in Earlier Revisions	4

5	Related Work	5
5.1	Content Moderation Systems	5
5.2	Guardrail Models	5
5.3	Multilingual Safety	5
6	Deployment Recommendations	5
6.1	Integration Patterns	5
6.2	Threshold Configuration	5
7	Conclusion	6

1 Introduction

Content safety classification is a critical infrastructure component for any deployed language model service. Effective moderation must satisfy competing constraints: high recall on harmful content (to prevent policy violations), high precision (to avoid false positives that degrade user experience), coverage across 100+ languages (to serve global user populations), and sub-10ms inference latency (to insert into API request paths without perceptible overhead).

Existing safety approaches trade off along several dimensions:

- Keyword filters: high false positive rate, trivially bypassed by paraphrasing
- Narrow binary classifiers: insufficient granularity for nuanced policy enforcement
- Large safety-tuned generation models: accurate but heavier to serve
- Embedding-similarity approaches: brittle to adversarial rephrasing

Rather than train a safety classifier from scratch, Zen3-Guard adopts IBM’s openly released Granite Guardian 8B [4], which directly targets these trade-offs as a purpose-built guardrail LLM. Our contribution is integration and packaging, not modeling. The upstream model provides:

1. **A risk-conditioned guardrail LLM** — A dense Granite 3.x 8B Instruct model fine-tuned by IBM to classify content against a named risk definition supplied at inference time, emitting a **Yes/No** verdict plus a harm probability.
2. **An IBM AI Risk Atlas taxonomy** — Risk dimensions covering harmful content and (in the RAG setting) faithfulness risks, rather than a single binary safe/unsafe label.
3. **Prompt- and response-side checking** — The same model can be applied to the user input, the model output, or a RAG response against its retrieved context.
4. **Risk-definition prompting** — Because the desired risk is described in the prompt template at inference time, the deployed model can be steered toward different checks without retraining.

We do not modify the Granite Guardian weights; Zen3-Guard is the upstream model wired into the Zen serving stack. The remainder of this document describes that integration and summarizes IBM’s published design and results with attribution.

2 Architecture

2.1 Upstream Model: Granite Guardian 8B

Zen3-Guard is the IBM Granite Guardian 8B model served unmodified. Per IBM’s documentation [4], the model is a *dense, decoder-only* transformer (**GraniteForCausalLM**; GQA, RoPE, SwiGLU MLP, RMSNorm). It is *not* a mixture-of-experts model and has no notion of “expert clusters” or safety-specific routing. It is produced by supervised fine-tuning of the IBM Granite 3.x 8B Instruct base model on risk-annotated data. We make no architectural changes; earlier revisions of this document incorrectly described a bespoke “mixture-of-distilled-experts” backbone, which does not correspond to the deployed weights.

2.2 Risk Taxonomy and Output Schema

Granite Guardian does not emit a fixed multi-label vector. Instead, the caller supplies a *risk definition* in the prompt template, and the model judges whether the supplied content exhibits that risk. The risk dimensions documented by IBM [4], aligned with the IBM AI Risk Atlas, include:

- **Harmful content:** harm (general), social bias, profanity, sexual content, unethical behavior, violence, and jailbreaking.
- **RAG faithfulness:** context relevance, groundedness, and answer relevance (i.e., detecting ungrounded or off-topic answers).

The IBM AI Risk Atlas mapping organizes harm into a deeper hierarchy (four high-level categories and thirteen sub-categories), but the model’s runtime interface is per-risk rather than a flat 47-way classifier.

For the selected risk, the model produces:

- Binary verdict: a **Yes** / **No** token (**Yes** = risk present).
- Probability of harm: derived from the log-probabilities of the **Yes/No** tokens at the final position.

Severity tiers and character-offset evidence spans are not native outputs of the upstream model; where downstream Zen tooling needs them, they are derived heuristically outside the model and are not part of Granite Guardian’s reported behavior.

2.3 Language Coverage

Granite Guardian inherits the tokenizer and multilingual coverage of its Granite 3.x base. IBM’s published evaluations focus primarily on English safety benchmarks; we therefore do not assert specific per-language safety scores. Claims in earlier revisions of a custom 150,528-token multilingual tokenizer and a cross-lingual transfer objective were fabricated and have been removed.

2.4 Risk-Definition Prompting

Because the risk to be checked is described in the chat template rather than baked into a classification head, a single deployed instance can perform different checks (e.g. `harm`, `social_bias`, `groundedness`) by changing the `risk_name` passed to the tokenizer’s `apply_chat_template`. Schematically:

```
apply_chat_template(  
    messages=[{"role": "user", "content": <user content>},  
              {"role": "assistant", "content": <response>}],  
    guardian_config={"risk_name": "harm"})  
# -> model emits "Yes"/"No"; prob_of_risk read from token logits
```

This is the upstream interface; Zen deployments expose it through a thin wrapper. It is a prompt-time mechanism, not a learned “policy enforcement expert cluster.”

3 Training

3.1 Provenance, Not Training

Zen3-Guard performs no training of its own. The guardrail was trained by IBM Research; we summarize their published methodology for completeness and attribute it accordingly [4].

Per IBM, Granite Guardian is created by *supervised fine-tuning* of the Granite 3.x 8B Instruct base model on a combination of human-annotated and synthetic risk data spanning the harm and RAG-faithfulness dimensions of the IBM AI Risk Atlas. The objective is standard next-token supervision over the **Yes/No** verdict given a risk-conditioned prompt; the harm probability used downstream is read from the **Yes/No** token log-probabilities at inference time.

The fabricated training pipeline in earlier revisions of this document — a 12B-token multilingual safety corpus, focal-loss “safety expert” specialization, a five-level taxonomy hierarchy loss, and a four-phase adversarial regimen — did not describe any real training run and has been removed. For the authoritative dataset composition and training recipe, see IBM’s paper [4].

3.2 Calibration

We do not recalibrate the model. IBM reports its evaluation metrics (Section 4) at a default decision threshold of 0.5 on the harm probability; deployments may pick a different threshold per risk to trade recall against false positives.

4 Evaluation

All numbers in this section are IBM’s published results for Granite Guardian 3.0 8B [4], reproduced here with attribution. We ran no evaluations of our own and make no independent benchmark claims.

4.1 Aggregate Results Reported by IBM

IBM reports two headline metrics for Granite Guardian 3.0 8B, aggregated over public benchmark suites:

Table 1: Granite Guardian 3.0 8B aggregate metrics, as reported by IBM [4].

Capability	Metric	Value
Harmful-content detection	Aggregate AUC	0.871
Harmful-content detection	Aggregate F1 (threshold 0.5)	0.758
RAG hallucination / groundedness	Average AUC (TRUE suite)	0.854

The harmful-content figures are aggregated across public safety datasets used in IBM’s evaluation (for example ToxicChat, HarmBench, SafeRLHF, BeaverTails, OpenAI Moderation, SimpleSafetyTests, and xstest), with comparisons against Llama Guard and ShieldGemma baselines; the RAG-hallucination figure is the average AUC on the TRUE benchmark [4]. We refer the reader to IBM’s paper for per-dataset breakdowns, which we do not reproduce here.

4.2 A Note on the Numbers in Earlier Revisions

Earlier revisions of this document reported figures such as ToxiGen 99.2%, HatEval 97.8%, a multilingual macro-F1 of 0.968, sub-5ms A10G latency, and per-attack adversarial robustness rates. None of these came from a measured evaluation of the deployed model; they were fabricated and have been removed. Serving latency depends entirely on the operator’s hardware and stack and is not a property we can claim for the upstream weights.

5 Related Work

5.1 Content Moderation Systems

Early content moderation relied on expert-curated keyword blocklists and regular expression patterns. Machine learning approaches improved recall but suffered from adversarial brittleness. Transformer-based classifiers [1] brought contextual understanding but required separate models per language.

5.2 Guardrail Models

Constitutional AI [2] demonstrated that safety behaviors can be instilled in models through targeted fine-tuning. A subsequent line of work — including Llama Guard, ShieldGemma, and IBM’s Granite Guardian [4] — packages a fine-tuned LLM specifically as an input/output guardrail that emits a risk verdict. Zen3-Guard does not introduce a new model in this line; it adopts Granite Guardian directly.

5.3 Multilingual Safety

Cross-lingual safety classification has been studied through multilingual pre-training and transfer [3]. We make no multilingual-safety claims for Zen3-Guard beyond what IBM reports for Granite Guardian, whose published evaluations are primarily English; the cross-lingual “alignment mechanism” described in earlier revisions did not exist and has been removed.

6 Deployment Recommendations

6.1 Integration Patterns

Zen3-Guard is designed for three primary deployment patterns:

Synchronous API guard: Insert as a pre-processing layer before the main LLM, classifying the user prompt and rejecting or flagging it before generation.

Parallel screening: Run the guard concurrently with the main LLM on the input, and suppress responses if it returns a positive risk verdict. On GPU clusters with spare capacity this can overlap with generation.

Output scanning: Run the guard on model outputs (and, in RAG, against the retrieved context for groundedness) before returning to the user. Recommended where harmful or ungrounded content may emerge from a benign-looking input.

Because the guard is itself an 8B decoder LLM, its serving cost is comparable to running an 8B model; we make no specific latency guarantee, as it depends on the operator’s hardware, batching, and quantization.

6.2 Threshold Configuration

The harm probability is read from the **Yes/No** token log-probabilities. IBM reports its metrics at a default threshold of $p = 0.5$ [4]. Operators may raise the threshold per risk to favor precision (fewer false positives) or lower it to favor recall (catching more borderline content); the appropriate operating point is deployment-specific and should be chosen against the operator’s own labeled data rather than the illustrative recall/FPR figures asserted in earlier revisions, which were not measured.

7 Conclusion

Zen3-Guard is the content-safety layer of the Zen model family, implemented as a direct deployment of IBM’s Apache-2.0 Granite Guardian 8B guardrail [4]. It is a dense, decoder-only Granite 3.x 8B Instruct model fine-tuned by IBM Research to classify prompts and responses against a named risk and emit a **Yes/No** verdict with a harm probability. IBM reports an aggregate AUC of 0.871 on harmful-content benchmarks and 0.854 on RAG-hallucination benchmarks for the 3.0 8B model. We contribute integration and packaging into the Zen serving stack, not a new model, and we make no benchmark claims of our own. Operators evaluating Zen3-Guard for production should validate it on their own content distribution and choose per-risk thresholds accordingly.

Acknowledgments and Attribution

Zen3-Guard is built entirely on IBM Granite Guardian ([ibm-granite/granite-guardian](#), Apache-2.0), developed by IBM Research; all model weights, training data, and reported metrics are IBM’s. We thank the IBM Granite team for releasing the model openly.

References

- [1] E. Lees et al., “A New Generation of Perspective API: Efficient Multilingual Character-level Transformers,” *ACL*, 2022.
- [2] Y. Bai et al., “Constitutional AI: Harmlessness from AI Feedback,” *arXiv:2212.08073*, 2022.
- [3] T. Hartvigsen et al., “ToxiGen: A Large-Scale Machine-Generated Dataset for Adversarial and Implicit Hate Speech Detection,” *ACL*, 2022.
- [4] I. Padhi et al. (IBM Research), “Granite Guardian,” *arXiv:2412.07724*, 2024. Model: [ibm-granite/granite-guardian-3.0-8b](#) (Apache-2.0).