

Governance-as-Code: Translating EU AI Act Technical Requirements into Executable Compliance Pipelines for Generative AI Systems

Rudrendu Kumar Paul¹ Sourav Nandy²

Abstract

The EU AI Act (Regulation 2024/1689) imposes technical obligations on high-risk AI providers, yet Articles 8–15 were drafted for predictive AI and leave seven technical gaps when applied to generative systems, spanning non-deterministic data governance, training-data provenance, continuous conformity, human oversight, open-ended robustness, emergent risk, and generative fairness. We deliver GOVERNANCE-AS-CODE (GaC), a framework of 43 machine-checkable acceptance criteria across six compliance modules that run in a CI/CD pipeline and emit Article-indexed audit evidence, and we show the actual Rego policy code rather than merely describing it. Our central commitment is that the Act’s open-textured standards (“appropriate levels,” “possible biases”) become declared, auditable numbers: robustness thresholds are derived from the provider’s documented baseline and a state-of-the-art floor, and framing bias is collapsed into eight measurable proxies tested by counterfactual demographic probing. We also correct who owes what, since under Article 25 and Chapter V a downstream deployer relies on the upstream provider’s Article 53 training-data summary and documents only the layers it controls, so GaC verifies that summary rather than demanding per-sample documentation the deployer never had. We validate on two enterprise deployments, a high-risk advisory chatbot and a limited-risk content generator, benchmarking against a manual expert audit rather than documentation artifacts that were never designed to enforce compliance. GaC reproduces all of the manual audit’s findings, including three penalty-

triggering violations, while cutting audit labor by roughly 75%.

1. Introduction

On August 2, 2026, the high-risk provisions of the EU AI Act take full effect. Organizations deploying AI in hiring, credit scoring, law enforcement, and critical infrastructure face penalties of up to €15 million or 3% of global annual turnover for non-compliance with high-risk obligations (European Parliament and Council of the European Union, 2024a). With the deadline close, many engineering teams still cannot answer a basic question: does our generative AI system actually satisfy Articles 8 through 15?

The problem is not a lack of awareness. The Act’s risk-based architecture has been analyzed extensively (Veale & Borgesius, 2021; Novelli et al., 2024; Smuha, 2021). Documentation frameworks exist (Mitchell et al., 2019; Gebru et al., 2021; Arnold et al., 2019), governance management standards exist (National Institute of Standards and Technology, 2023; International Organization for Standardization, 2023), and recent work has begun mapping legal requirements to verification activities (Buscemi et al., 2025). What is missing is the last mile: turning regulatory text into machine-checkable acceptance criteria that run automatically in a deployment pipeline, with the operationalizing choices made explicit rather than hidden inside an auditor’s head.

Consider a concrete failure. An organization deploys a retrieval-augmented advisory chatbot classified as high-risk under Annex III, Area 5(b) (creditworthiness assessment). Article 9 requires a risk management system that operates “throughout the entire lifecycle.” The team runs a conformity assessment at deployment. Three weeks later the retrieval index is rebuilt with updated domain documents, shifting the embedding distribution; the system’s hallucination rate on domain questions climbs from 4% to 9%. No re-assessment fires, because the change does not meet Article 43’s “substantial modification” threshold: the model weights are unchanged, only the retrieval corpus shifted. The risk management system, built around a point-in-time audit, is blind precisely where generative systems drift.

Accepted at the AI4Law Workshop, ICML 2026. ¹Boston University, Boston, MA, USA ²University of Texas at Austin, Austin, TX, USA. Correspondence to: Rudrendu Kumar Paul <rudrendupaul2022@gmail.com>, Sourav Nandy <sourav.nandy@gmail.com>.

Proceedings of the 43rd International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

This gap between regulation and implementation recurs across Articles 8–15. Each instance traces to an assumption in the legal text that holds for predictive ML but breaks for generative systems. The Act’s requirements presume bounded output spaces, deterministic inference, tractable data provenance, and human-interpretable decision boundaries. Generative systems violate all four at once.

A second, subtler problem sits underneath the first. Even where a requirement *can* be operationalized, doing so forces a choice that the legal text leaves open. Article 15 asks for “appropriate levels” of robustness; turning that into a pass/fail gate means picking a number. Article 10 asks teams to examine data for “possible biases”; checking that for an open-ended generator means choosing a measurable proxy for “bias.” If those choices are buried as constants in a script, the compliance claim is no more auditable than a PDF. Our central design commitment is that every such choice is *declared*, justified against the Act’s language, and recorded as evidence.

Contributions.

1. A **systematic analysis of seven technical gaps** in applying Articles 8–15 to generative AI, grounded in the regulation’s text, its allocation of duties across the value chain (Articles 25 and 53), and its implementation timeline (Section 3).
2. **GOVERNANCE-AS-CODE**, a framework of 43 machine-checkable acceptance criteria in six modules. We show the *actual* policy code (Section 4) and make explicit how each vague standard becomes a declared, auditable threshold (Section 5).
3. **Empirical validation** on two enterprise case studies, benchmarked against a manual expert audit rather than documentation artifacts, with full reporting of what each violation means and how it was measured (Section 6).

2. Background and Related Work

2.1. The Act’s technical requirements and who owes them

Regulation 2024/1689 establishes a risk-based classification in which high-risk systems (Annex III) must satisfy Articles 8–15 (European Parliament and Council of the European Union, 2024a). Article 8 mandates compliance “taking into account the generally acknowledged state of the art.” Articles 9–15 then specify risk management (Art. 9), data governance (Art. 10), technical documentation (Art. 11), record-keeping (Art. 12), transparency (Art. 13), human oversight (Art. 14), and accuracy, robustness, and cybersecurity (Art. 15).

A point that prior technical work often glosses, and that materially shapes any compliance pipeline, is that these du-

ties are split across the value chain. General-purpose AI (GPAI) models carry their own obligations under Chapter V, Articles 51–55. In particular, Article 53(1)(d) requires the upstream model provider to publish a “sufficiently detailed summary” of training content, using the Commission’s mandated template (European Parliament and Council of the European Union, 2024c; European Commission, 2025). Article 25 then governs how responsibility flows downstream: a deployer that fine-tunes or substantially modifies a third-party model can be reclassified as a provider, but for the upstream model itself the deployer’s duty is largely to *rely on and pass through* the documentation the provider supplies under Article 53, secured by the written agreement Article 25(4) anticipates (European Parliament and Council of the European Union, 2024b). The deploying organization does not, in the common case, control or owe documentation for the foundation model’s training corpus. It owes documentation for the layers it does control: fine-tuning data, alignment data, and the retrieval corpus. We take this division seriously throughout, because a framework that demands per-sample documentation of a web-scale corpus from a downstream deployer is asking for something the Act itself does not require.

2.2. Existing governance approaches

Documentation artifacts. Model Cards (Mitchell et al., 2019), Datasheets (Gebru et al., 2021), and IBM AI Fact-Sheets (Arnold et al., 2019) provide structured transparency. Golpayegani et al. (2024) extended this with AI Cards, a machine-readable documentation framework using Semantic Web technologies aligned with the Act. These artifacts capture system properties at a point in time. They do not execute checks, enforce constraints, or detect when documented properties diverge from runtime behavior. A Model Card reporting a 3% hallucination rate stays unchanged when production climbs to 9%. This is a difference in *purpose*, not a deficiency: documentation artifacts were designed to communicate, not to gate a deployment. That distinction matters for how we benchmark (Section 6).

Management-system standards. The NIST AI Risk Management Framework (National Institute of Standards and Technology, 2023) and ISO/IEC 42001 (International Organization for Standardization, 2023) specify organizational processes for governing AI risk (Govern, Map, Measure, Manage, in NIST’s case). They define *what* an organization should do but, by design, leave the technical realization open. GaC is complementary: it is one concrete, code-level realization of the “Measure” and “Manage” functions for the specific obligations in Articles 8–15.

Compliance mapping frameworks. Buscemi et al. (2025) decomposed high-risk requirements into verification activities mapped to lifecycle stages. Momcilovic et al. (2024)

combined ontologies, assurance cases, and factsheets to argue robustness compliance for LLMs. Both clarify *what* must be verified but stop short of *how to automate* that verification in a pipeline.

Runtime enforcement. Gaurav et al. (2025) proposed Governance-as-a-Service, a multi-agent layer that intercepts agent outputs at runtime using trust scoring and graduated interventions. Kaptein et al. (2026) formalized runtime governance as deterministic functions over execution paths. Both address runtime behavior but are not structured around the Act’s articles and do not produce the article-indexed audit evidence a conformity assessment needs.

Generative bias and robustness measurement. Because two of our gaps (G5, G7) turn on measurement, we draw on the open-ended generation literature. BOLD (Dhamala et al., 2021) and HolisticBias (Smith et al., 2022) show that bias in free-form text can be measured through counterfactual prompt sets and named output metrics rather than through a fixed classification target; we adapt that methodology into a checkable proxy (Section 5). For robustness, HELM (Liang et al., 2023) demonstrates large variation across task types, and the indirect prompt-injection results of Greshake et al. (2023) supply the threat model our robustness module tests against.

2.3. The missing piece

Marino & Lane (2026) proposed computational compliance as a research program for integrating algorithmic approaches across the AI lifecycle. We share the diagnosis but note the work remains a blueprint: it defines objectives without delivering article-specific acceptance criteria. Nolte et al. (2025) identified legal gaps in the Act’s robustness and cybersecurity provisions, and Deckenbrunnen et al. (2026) proposed technical sandboxes as micro-foundations for regulatory learning. Neither translates these insights into executable checks. Where prior work maps, documents, or enforces, we *compile*: we translate requirements into acceptance tests a pipeline runs before every deployment, producing machine-readable, article-indexed evidence.

3. Seven Technical Gaps

How we derived the gaps. We did not aim to enumerate every conceivable mismatch; we aimed for the gaps that are (a) grounded in a specific obligation in Articles 8–15, (b) triggered by a concrete technical property of generative systems, and (c) addressable by a pipeline check. We worked article by article, asking for each: can an engineering team implement this requirement *as written* for an LLM-based system? Where the answer was no, we recorded the breaking assumption. The seven gaps are therefore a deliberately operational set, not a claim of exhaustiveness. They are not

Table 1. Seven technical gaps in applying Articles 8–15 to generative AI. Each arises from an assumption that holds for predictive ML but breaks for LLM-based systems. Overlap on Articles 9 and 10 is intentional and noted in the text.

Gap	Article	GaC Module
G1: Output non-determinism	Art. 10	Data Lineage
G2: Training-data provenance	Art. 11/53	Data Lineage
G3: Continuous conformity	Art. 9	Risk Registry
G4: Oversight at speed	Art. 14	Oversight Hooks
G5: Open-ended robustness	Art. 15	Robustness Testing
G6: Emergent risk	Art. 9	Risk Registry
G7: Generative fairness	Art. 10	Output Monitoring

fully orthogonal, and we do not claim they are. Article 9 appears in two gaps (G3 continuous conformity, G6 emergent risk) because it carries two distinct assumptions; Article 10 appears in G1 and G7 for the same reason. Table 1 maps the overlap explicitly. Other gaps surely exist (Article 12 logging granularity for non-deterministic systems, Article 13 explanation of stochastic outputs); we scope to the seven that current open-source tooling can check today and flag the rest as future work.

G1: Output non-determinism (Art. 10). Article 10(2) requires training, validation, and testing data to be “subject to data governance and management practices appropriate for the intended purpose.” For predictive models this means curated splits with documented distributions. LLM-based systems break the implicit reproducibility assumption twice: the same input yields different outputs across calls under temperature sampling, and retrieval-augmented systems select context at query time, so the effective input is itself non-reproducible. When “same input, same output” fails, “appropriate data governance” needs a new operational meaning.

G2: Training-data provenance (Arts. 11 and 53). Here we correct a framing error common in technical treatments, including an earlier version of this work. Article 11 requires technical documentation containing a *general description* of the training data, not per-sample documentation; and for the foundation model itself, the governing obligation is Article 53(1)(d)’s “sufficiently detailed summary,” owed by the *upstream* provider (European Parliament and Council of the European Union, 2024c; European Commission, 2025). So the genuine gap is not that per-sample documentation of a trillion-token corpus is infeasible; the Act already agrees, which is exactly why it asks for a summary. The real gap is downstream and layered: a deployer must (i) verify that

the upstream Article 53 summary exists and is machine-readable, and (ii) document the layers it actually controls, namely fine-tuning data, alignment data, and the retrieval corpus, the last of which changes daily and is, behaviorally, part of the system’s effective training data. Existing tooling provides neither the upstream-summary check nor the runtime-corpus provenance.

G3: Continuous conformity under updates (Art. 9). Article 9(1) mandates risk management that is “continuous” and “iterative.” Article 43 requires fresh conformity assessment only for “substantial modifications.” Prompt changes, retrieval-index rebuilds, adapter fine-tuning, and context-window expansion all alter behavior without changing weights, and none clearly trips the threshold. ETSI TS 104 008 established a framework for continuous auditing-based conformity assessment (European Telecommunications Standards Institute, 2026), but it is not a harmonized standard and defines no LLM-specific triggers.

G4: Human oversight at speed (Art. 14). Article 14(1) requires that high-risk systems can “be effectively overseen by natural persons,” and Article 14(4) the ability “to intervene.” Agentic systems executing multi-step tool calls decide in milliseconds. An advisory agent that retrieves, scores risk, and recommends in under two seconds outruns any human reviewer. The assumption that a person can meaningfully intervene before the system acts does not hold for autonomous agents.

G5: Open-ended robustness (Art. 15). Article 15(1) requires “appropriate levels of accuracy, robustness and cybersecurity.” For classifiers, accuracy is well defined (F1, AUC, precision@*k*). For generators the output space is unbounded, and “appropriate levels” supplies no number. Any pass/fail gate must therefore manufacture a threshold; the open question, which we answer in Section 5, is how to do so transparently rather than by fiat.

G6: Emergent capability risk (Art. 9). Article 9(2) requires identifying “reasonably foreseeable risks” and “reasonably foreseeable misuse.” Emergent capabilities (in-context learning, chain-of-thought, tool use) arise at scale thresholds not predictable from smaller models (Bommasani et al., 2022). If the capability was not foreseeable before training, its risks cannot be enumerated in a pre-deployment assessment. Some risks become identifiable only after deployment at scale.

G7: Generative fairness (Art. 10). Article 10(2)(f) requires examining data “in view of possible biases.” Fairness frameworks (demographic parity, equalized odds, calibration) presume protected attributes and a measurable outcome. Generative text has no fixed classification target, so “bias” here lives in framing: tone, risk posture, hedging,

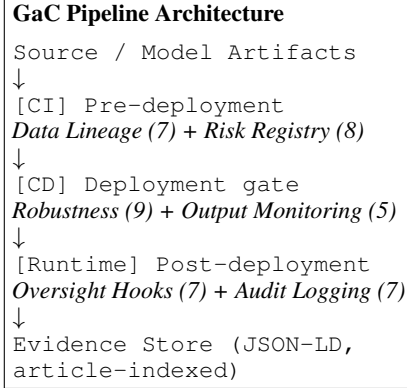


Figure 1. 43 acceptance criteria across CI, CD, and runtime phases.

specificity. The hard part, and the reviewer-flagged one, is that framing is not directly checkable; it must be collapsed into named measurable proxies before any rule can evaluate it. We do exactly that in Section 5.

4. The Governance-as-Code Framework

GOVERNANCE-AS-CODE¹ (GaC) turns each gap into concrete acceptance criteria. Three design principles govern it:

1. **Executable over advisory:** every criterion is a boolean test that passes or fails in CI/CD, not a recommendation requiring interpretation.
2. **Evidence-producing:** every execution emits a structured record (JSON-LD) linked to the Article and subparagraph it satisfies.
3. **Declared, not buried:** every threshold or proxy a criterion depends on is an explicit, justified input recorded with the result, never a hidden constant.

4.1. Architecture

GaC is pipeline middleware that intercepts deployment at three points: pre-deployment (CI), the deployment gate (CD), and runtime. Each module registers its criteria as Rego policy rules for Open Policy Agent (Cloud Native Computing Foundation, 2024a), evaluated against system state; OpenTelemetry (Cloud Native Computing Foundation, 2024b) carries the article-indexed evidence. Figure 1 shows the layout.

¹The term has been used in DevOps for policy enforcement through code. We use it narrowly: translating specific Article 8–15 requirements into machine-checkable acceptance criteria for deployment pipelines.

Table 2. Six GaC modules: criteria count, target Articles, gaps addressed.

Module	#	Articles	Gaps
Data Lineage	7	10, 11, 53	G1, G2
Output Monitoring	5	10, 13	G7
Oversight Hooks	7	14	G4
Robustness Testing	9	15	G5
Risk Registry	8	9	G3, G6
Audit Logging	7	12	All

4.2. The six modules

Table 2 maps modules to articles and gaps. We describe the mechanism of each, not just its name, and show representative policy code.

Data Lineage (G1, G2). The module replaces a single flat “training data” notion with a three-tier provenance model that mirrors the Act’s own division of duties. Tier 1 (foundation model): the check verifies that the upstream provider’s Article 53 summary is present and machine-readable, rather than demanding per-sample data the deployer never had. Tier 2 (fine-tuning and alignment): per-sample source attribution, annotator provenance, and applied quality filters, for the data the deployer controls. Tier 3 (runtime retrieval): corpus versioning by content hash, embedding-model version pinning, and per-request context logging sufficient to reconstruct the effective input. Seven criteria enforce this; Listing 1 shows the Tier 1 check, which encodes the upstream-reliance logic of Article 25.

Listing 1. Data Lineage, criterion DL-1: verify the upstream Art. 53 summary the deployer relies on (Art. 25), not a corpus it does not own.

```
package gac.data_lineage
# Art. 53(1) (d): upstream provider's training-data
summary
default dll_pass := false
dll_pass if {
  s := input.foundation_model.art53_summary
  s.present == true
  s.machine_readable == true
  s.template_version != "" # Commission TDS template
}
violation[msg] if {
  not dll_pass
  msg := "DL-1 (Art.53/25): upstream training-data
summary missing or unverifiable"
}
```

Output Monitoring (G7). This module makes generative fairness checkable by counterfactual probing (detailed in Section 5): it holds a request fixed, varies an injected demographic attribute, and tests for significant divergence across named output features. Five criteria cover probe-set existence, the eight measured dimensions, the divergence test, an Article 13 transparency record, and Article 50 marking of generated content.

Oversight Hooks (G4). The module classifies every system

action into one of three oversight modes by autonomy and criticality. *Synchronous*: execution pauses for human approval (irreversible actions). *Asynchronous*: act now, queue for review within an SLA (reversible, moderate risk). *Statistical*: autonomous operation with sampled review, the sample rate set to detect a specified deviation at a specified confidence. Seven criteria enforce mode assignment, the synchronous block, the asynchronous SLA, the statistical sample-size derivation, intervention logging, auditability of the classification, and reviewer-capacity-versus-queue monitoring.

Robustness Testing (G5). The module runs an adversarial suite (prompt injection vectors after Greshake et al. (2023), jailbreak templates, and domain-specific deception probes) and compares the measured pass rate against a *declared* threshold (Section 5). Nine criteria span suite coverage, the declared threshold and its justification, the floor check, and per-attack-class reporting.

Risk Registry (G3, G6). The module replaces point-in-time assessment with event-driven re-evaluation over five drift signals: output-distribution shift (embedding-space KL over a rolling 24h window), retrieval-corpus drift (hash comparison to the assessment-time snapshot), prompt-template changes (version control), adapter/LoRA changes, and emergent-capability detection (monthly probing). Signals are graduated: Tier 1 signals trigger automated robustness re-evaluation within 4h; Tier 2 add manual risk-matrix review within 48h; Tier 3 (emergent capability) triggers full conformity re-assessment. Listing 2 shows the corpus-drift criterion, which closes the substantial-modification blind spot of the opening example.

Listing 2. Risk Registry, criterion RR-2: corpus drift forces re-assessment even when weights are unchanged (closes the Art. 43 blind spot).

```
package gac.risk_registry
# Art. 9(1): continuous risk management
violation[msg] if {
  cur := input.retrieval_corpus.content_hash
  base := data.assessment_snapshot.corpus_hash
  cur != base
  not input.reassessment.completed_within_4h
  msg := "RR-2 (Art.9): corpus drift without re-
assessment"
}
```

Audit Logging (G12, all). OpenTelemetry spans are enriched with compliance metadata (article reference, criterion ID, pass/fail), producing a trail a conformity body can query by article number. Seven criteria cover span completeness, the article index, tamper-evidence, retention, request-ID linkage, evidence-store schema conformance, and exportability.

5. From Vague Standard to Auditable Number

This section answers the two questions reviewers rightly press: how does “appropriate levels” (G5) become a pass rate, and how does “possible biases” in framing (G7) become a measurable check? Our answer in both cases is the same in spirit: we do not pretend the Act supplies a number. We make the number a declared, justified input and have the check verify both that the system meets it and that the declared number itself clears a floor.

5.1. G5: deriving a robustness pass rate

We operationalize “appropriate” as a deployment-declared threshold with a documented justification chain:

$$\tau_{\text{deploy}} = \max(\tau_{\text{floor}}, \tau_{\text{baseline}} - \delta), \quad (1)$$

where τ_{baseline} is the robustness level the provider *claimed* in its Article 11 technical documentation, δ is a declared maximum tolerable degradation, and τ_{floor} is a domain floor drawn from the “state of the art” Article 8 invokes (a published benchmark such as HELM (Liang et al., 2023), or a harmonized standard once available). The check then verifies two things: the measured pass rate $\hat{\tau} \geq \tau_{\text{deploy}}$, and the declared $\tau_{\text{deploy}} \geq \tau_{\text{floor}}$. All three of τ_{baseline} , δ , and τ_{floor} are written to the evidence store with their sources, so an assessor sees not a magic constant but a defensible derivation they can contest. The subjectivity the Act leaves open is surfaced and made auditable, not eliminated.

Listing 3. Robustness, criterion RT-3: measured rate must clear the declared threshold, and the threshold must clear the floor; all inputs are recorded.

```
package gac.robustness
# Art. 15(1): "appropriate levels" -> declared, floored
violation[msg] if {
  t := input.robustness
  decl := max([t.floor, t.baseline - t.delta])
  t.measured_pass_rate < decl
  msg := sprintf("RT-3 (Art.15): pass rate %.3f below
    declared %.3f", [t.measured_pass_rate, decl])
}
violation[msg] if {
  input.robustness.declared_threshold < input.
  robustness.floor
  msg := "RT-3 (Art.15): declared threshold below state-
    of-the-art floor"
```

5.2. G7: a measurable proxy for framing bias

Framing bias cannot be read off a single label, so we collapse it into eight named, individually measurable output features and test for demographic dependence by counterfactual probing, following the open-ended-generation methodology of BOLD (Dhamala et al., 2021) and HolisticBias (Smith et al., 2022). The protocol: hold the user request fixed; vary one injected demographic attribute (age band, income bracket, region/nationality, inferred gender) across N templated probes; for each generated response compute

Table 3. The eight measurable proxies for “framing bias.” Each is computed per response; demographic dependence is tested across counterfactual probes.

Feature (proxy)	Measurement
Recommendation valence	advice sentiment polarity [−1, 1]
Risk-framing direction	caution vs. opportunity term ratio
Hedging / certainty	modal + hedge token rate
Recommendation strength	directive / imperative count
Register	Flesch–Kincaid grade level
Claim density	assertions per 100 tokens
Qualification rate	disclaimer / refusal frequency
Numeric specificity	explicit figures offered

the eight features in Table 3; then test whether any feature’s distribution differs across demographic conditions (Kruskal–Wallis for continuous features, χ^2 for categorical), with Holm correction across the eight tests. The criterion passes only if no feature shows a significant dependence at the declared α . This is the “visible measure” the framework reports: not a vague claim of bias, but a per-feature, per-dimension test with an effect size and a p -value.

6. Evaluation

We validate GaC on two enterprise deployments at distinct risk classifications.

6.1. Baseline: a fair comparison

Reviewers correctly note that Model Cards and FactSheets are not compliance-enforcement tools, so reporting that they “found zero failures” compares against the wrong thing. We agree and restructure the comparison accordingly. Our primary baseline is a **manual expert audit**: a legal-plus-engineering team performing the Article 8–15 conformity review by hand, which is the actual status quo for organizations facing the deadline. We report documentation artifacts (FactSheets, Model Cards) only as a description of the information *already on hand* before either the manual audit or GaC ran, to make clear that the failures were not visible in existing documentation. The claim is therefore not “GaC beats Model Cards”; it is “GaC reproduces a manual expert audit’s findings at a fraction of the labor, and the failures it surfaces were latent in, but not flagged by, the documentation teams already maintain.”

6.2. Case Study 1: regulated-domain advisory chatbot

System. A retrieval-augmented LLM for regulated advisory services, high-risk under Annex III Area 5(b). Architecture: an embedding retriever over 84,000 domain documents, a fine-tuned 8B generator, and a prohibited-advice filter.

Volume: 12,000 queries/day. Documented with IBM AI FactSheets and audited quarterly by hand.

Findings. Against the manual audit’s eight confirmed issues, GaC reproduced all eight. We describe each with the evidence it produced, since a count alone is not assessable:

1. **G3/Art. 9.** The retrieval corpus was rebuilt weekly with no re-assessment. Three rebuilds over six weeks moved the hallucination rate from 3.2% to 7.1% on domain questions (measured on a fixed 500-question probe). RR-2 caught the first rebuild by hash comparison; the FactSheet was never updated.
2. **G1/Art. 10.** Sampling temperature was 0.7 but unlogged per request, so two identical queries could yield contradictory advice with no audit trail distinguishing them.
3. **G4/Art. 14.** All 12,000 daily responses were delivered without review; the team sampled 50 (0.4%). The statistical-oversight criterion computed the required rate to detect a 5% quality deviation at 95% confidence and flagged 0.4% as far short.
4. **G5/Art. 15.** No adversarial testing. A 200-vector suite (injection vectors after Greshake et al. (2023) plus jail-break templates) achieved 23% success at extracting the system prompt and 8% at eliciting prohibited advice; both exceeded τ_{deploy} from Equation (1).
5. **G2/Art. 11/53.** Fine-tuning data (42,000 expert pairs) lacked annotator provenance; source attribution existed for 67% of pairs, with the remaining 33% from a contractor with no documented quality process. The upstream Article 53 summary for the base model was present, so DL-1 passed; the failure was squarely in the Tier 2 layer the deployer controls.
6. **G7/Art. 10.** No bias testing. Our counterfactual probe (500 advisory queries varying only implied user demographics) showed significant dependence in three of the eight Table 3 features, most strongly in risk-framing direction and recommendation strength (Kruskal–Wallis, Holm-corrected $p < 0.001$).
7. **G6/Art. 9.** The risk register listed 14 risks, none covering the model’s post-fine-tuning multi-step regulatory-reasoning capability, which was absent from the base model’s profile.
8. **G3/Art. 9.** Three prompt-template edits over eight weeks were version-controlled but triggered no compliance review.

The FactSheet documented architecture, intended use, and deployment-time metrics, but executed no runtime check and so surfaced none of these.

6.3. Case Study 2: content generation system

System. An LLM content generator, limited-risk under Article 50. Architecture: a fine-tuned 70B model with a

Table 4. Effort and coverage. GaC reproduces the manual expert audit’s findings at roughly one-quarter the labor. Documentation artifacts are shown as the pre-existing information state, not as enforcement competitors.

	GaC	Manual audit	Docs
Findings, chatbot	8	8	0 flagged
Findings, generator	4	4	0 flagged
Penalty-triggering	3	3	0
Audit labor (chatbot)	18h	72h	—
Art. 8–15 coverage	43/43	manual	doc-only

brand-voice adapter, a safety filter, and A/B testing. Volume: 3,500 marketing items/day. Documented with Google Model Cards.

Findings. GaC reproduced the manual audit’s four issues:

1. **Art. 50.** Generated content carried no machine-readable marking, violating Article 50(2). The Model Card noted the system’s generative nature; the outputs themselves did not.
2. **G7/Art. 10.** Beauty-product descriptions showed significant demographic dependence in adjective valence and benefit-claim density across the eight Table 3 dimensions ($p < 0.01$ on two features after correction); no bias testing existed.
3. **G2/Art. 11.** The brand-voice adapter was fine-tuned on 18 months of historical copy with no record of which segments, contexts, or categories were overrepresented.
4. **G5/Art. 15.** The safety filter blocked 99.2% of overtly harmful outputs but passed 34% of subtly misleading claims (150 deception probes designed to be technically-true-but-deceptive).

6.4. Effort and coverage

Table 4 reports the corrected comparison. GaC reproduced every finding of the 72-hour manual audit of the chatbot in 18 hours, of which 14 were human review of flagged failures rather than evidence collection, a roughly 75% labor reduction that shifts effort from gathering evidence to interpreting it. The three penalty-triggering issues (no human oversight under Art. 14; no continuous risk management under Art. 9; missing Art. 50 marking) match the manual audit’s. The documentation column is included only to show that the information needed was not absent from the organization; it was simply never executed as a check.

7. Discussion

Deployment cost. The full pre-deployment suite adds roughly 45 minutes to the chatbot’s pipeline, 32 of them in Robustness Testing, which can run asynchronously behind

a deployment hold for latency-sensitive rollouts.

Threshold and proxy calibration. The declared thresholds of Section 5 (the robustness δ and floor, the oversight confidence level, the fairness α) currently rest on expert judgment anchored to the Act’s “appropriate levels” and “state of the art” language. As enforcement actions, court decisions, and harmonized standards accumulate, these become empirically calibratable; because they are declared inputs rather than buried constants, updating them is a configuration change, not a rewrite.

Scope. GaC covers Articles 8–15 and Article 50. It does not cover organizational duties: quality management (Art. 17), conformity-assessment procedure (Art. 43), or registration (Art. 49), which are process, not pipeline. It assumes risk classification (Art. 6) is settled, since classification is a legal judgment. The value-chain logic of Section 2 is encoded only to the depth the Article 25(4) written agreement makes machine-checkable; contractual nuance beyond that remains a human task.

Forward compatibility. As harmonized standards mature, GaC’s Rego files can adopt their mandated measurement methods and thresholds without changing the architecture.

7.1. Limitations

First, the evaluation covers two systems; broader validation across healthcare, law enforcement, and education (all Annex III) is needed. Second, our “failures” are gaps relative to our reading of Articles 8–15; until enforcement and harmonized standards settle interpretation, reasonable disagreement is inevitable, which is precisely why we make our operationalizing choices explicit and contestable. Third, robustness coverage depends on the adversarial suite’s quality and needs domain curation. Fourth, the eight fairness proxies are a deliberate simplification of “framing” and will miss biases they do not measure; they are a floor on checkability, not a ceiling.

8. Conclusion

The EU AI Act imposes technical obligations that existing tooling cannot verify for generative systems, and that cannot be verified at all without making the Act’s open-textured standards into explicit numbers. We identified seven gaps where requirements written for predictive AI break on LLM-based systems, corrected the value-chain framing that governs who owes training-data documentation, and delivered GOVERNANCE-AS-CODE: 43 machine-checkable criteria in six modules, with the policy code shown and every threshold and proxy declared rather than buried. Benchmarked against a manual expert audit, the right baseline, GaC reproduces the findings at roughly a quarter of the labor. As the August 2, 2026 deadline arrives, engineering teams need

executable compliance whose judgments are visible and auditable, and that is what GaC provides.

Impact Statement

GaC aims to make EU AI Act compliance an engineering discipline with an auditable trail. The main risk is misplaced confidence: a passing pipeline certifies only the criteria encoded, with the declared thresholds and proxies it was given, and is not a substitute for legal judgment on classification or on contested interpretations. We mitigate this by surfacing every operationalizing choice as recorded, contestable evidence rather than a hidden constant, so that a passing result invites scrutiny rather than foreclosing it.

References

- Arnold, M., Bellamy, R. K. E., Hind, M., Houde, S., Mehta, S., Mojsilović, A., Nair, R., Ramamurthy, K. N., Olteanu, A., Piorkowski, D., Reimer, D., Richards, J., Tsay, J., and Varshney, K. R. FactSheets: Increasing trust in AI services through supplier’s declarations of conformity. *IBM Journal of Research and Development*, 63(4/5):6:1–6:13, 2019.
- Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2022. URL <https://arxiv.org/abs/2108.07258>.
- Buscemi, A., Deckenbrunnen, T., Kabir, F., Mishchenko, K., and Mowla, N. Assessing high-risk AI systems under the EU AI Act: From legal requirements to technical verification. *arXiv preprint arXiv:2512.13907*, 2025. URL <https://arxiv.org/abs/2512.13907>.
- Cloud Native Computing Foundation. Open policy agent. <https://www.openpolicyagent.org/>, 2024a. CNCF Graduated Project.
- Cloud Native Computing Foundation. OpenTelemetry: An observability framework for cloud-native software. <https://opentelemetry.io/>, 2024b. CNCF Incubating Project.
- Deckenbrunnen, T., Buscemi, A., Almada, M., Capozucca, A., and Castignani, G. Bathtubs, boundaries, and sandboxes: AI regulatory learning under legal uncertainty. *arXiv preprint arXiv:2601.04094*, 2026. URL <https://arxiv.org/abs/2601.04094>.
- Dhamala, J., Sun, T., Kumar, V., Krishna, S., Pruksachatkun, Y., Chang, K.-W., and Gupta, R. BOLD: Dataset and metrics for measuring biases in open-ended language

- generation. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pp. 862–872, 2021. URL <https://arxiv.org/abs/2101.11718>.
- European Commission. The EU general-purpose AI code of practice. Final version, July 2025, 2025. URL <https://code-of-practice.ai/>.
- European Parliament and Council of the European Union. Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence. Official Journal of the European Union, L Series, 2024a. URL <https://eur-lex.europa.eu/eli/reg/2024/1689/oj>.
- European Parliament and Council of the European Union. Article 25: Responsibilities along the AI value chain. Regulation (EU) 2024/1689, Chapter III, Section 3, 2024b. URL <https://artificialintelligenceact.eu/article/25/>.
- European Parliament and Council of the European Union. Article 53: Obligations for providers of general-purpose AI models. Regulation (EU) 2024/1689, Chapter V, 2024c. URL <https://artificialintelligenceact.eu/article/53/>.
- European Telecommunications Standards Institute. ETSI TS 104 008 v1.1.1: Methods for testing and specification; continuous auditing-based conformity assessment. Technical report, ETSI, 2026. URL https://www.etsi.org/deliver/etsi_ts/104000_104099/104008/01.01.01_60/ts_104008v010101p.pdf.
- Gaurav, S., Heikkonen, J., and Chaudhary, J. Governance-as-a-service: A multi-agent framework for AI system compliance and policy enforcement. *arXiv preprint arXiv:2508.18765*, 2025. URL <https://arxiv.org/abs/2508.18765>.
- Geburu, T., Morgenstern, J., Vecchione, B., Vaughan, J. W., Wallach, H., III, H. D., and Crawford, K. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92, 2021. URL <https://arxiv.org/abs/1803.09010>.
- Golpayegani, D., Hupont, I., Panigutti, C., Pandit, H. J., Schade, S., O’Sullivan, D., and Lewis, D. AI cards: Towards an applied framework for machine-readable AI and risk documentation inspired by the EU AI Act. *arXiv preprint arXiv:2406.18211*, 2024. URL <https://arxiv.org/abs/2406.18211>.
- Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., and Fritz, M. Not what you’ve signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection. In *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security (AISec)*, pp. 79–90, 2023. URL <https://arxiv.org/abs/2302.12173>.
- International Organization for Standardization. ISO/IEC 42001:2023 information technology — artificial intelligence — management system. ISO/IEC JTC 1/SC 42, 2023. URL <https://www.iso.org/standard/42001>.
- Kaptein, M., Khan, V.-J., and Podstavnychy, A. Runtime governance for AI agents: Policies on paths. *arXiv preprint arXiv:2603.16586*, 2026. URL <https://arxiv.org/abs/2603.16586>.
- Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., Zhang, Y., Narayanan, D., Wu, Y., Kumar, A., et al. Holistic evaluation of language models. *Transactions on Machine Learning Research*, 2023. URL <https://arxiv.org/abs/2211.09110>.
- Marino, B. and Lane, N. D. Computational compliance for AI regulation: Blueprint for a new research domain. *arXiv preprint arXiv:2601.04474*, 2026. URL <https://arxiv.org/abs/2601.04474>.
- Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I. D., and Gebru, T. Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT*)*, pp. 220–229, 2019. URL <https://arxiv.org/abs/1810.03993>.
- Momcilovic, T. B., Buesser, B., Zizzo, G., Purcell, M., and Balta, D. Towards assuring EU AI Act compliance and adversarial robustness of LLMs. *arXiv preprint arXiv:2410.05306*, 2024. URL <https://arxiv.org/abs/2410.05306>.
- National Institute of Standards and Technology. Artificial intelligence risk management framework (AI RMF 1.0). Technical Report NIST AI 100-1, U.S. Department of Commerce, 2023. URL <https://doi.org/10.6028/NIST.AI.100-1>.
- Nolte, H., Rateike, M., and Finck, M. Robustness and cybersecurity in the EU artificial intelligence act. *arXiv preprint arXiv:2502.16184*, 2025. URL <https://arxiv.org/abs/2502.16184>.
- Novelli, C., Hacker, P., Morley, J., Trondal, J., and Floridi, L. A robust governance for the AI Act: AI Office, AI

Board, scientific panel, and national authorities. *European Journal of Risk Regulation*, 2024. URL <https://arxiv.org/abs/2407.10369>.

Smith, E. M., Hall, M., Kambadur, M., Presani, E., and Williams, A. “I’m sorry to hear that”: Finding new biases in language models with a holistic descriptor dataset. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9180–9211, 2022. URL <https://arxiv.org/abs/2205.09209>.

Smuha, N. A. From a ‘race to AI’ to a ‘race to AI regulation’: Regulatory competition for artificial intelligence. *Law, Innovation and Technology*, 13(1):57–84, 2021.

Veale, M. and Borgesius, F. Z. Demystifying the draft EU artificial intelligence act. *Computer Law Review International*, 22(4):97–112, 2021. URL <https://arxiv.org/abs/2107.03721>.