

Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication

Soham S. Jadhav ^{1,✉}, Omkar N. Gadakh ¹, Atharv Gaikwad ¹, Nisha D. Patil ¹

1. Artificial Intelligence and Data Science, MET's Institute of Engineering, Nashik, IND

Received: July 03, 2026 | Review began: July 26, 2026 | Review ended: August 15, 2026 | Published: September 09, 2026

© Copyright 2026

This is an open access article distributed under the terms of the Creative Commons Attribution License CC-BY 4.0., which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract

Enterprise communication platforms carry text, image, and audio content across organizational contexts that public-facing moderation systems were never designed to serve. Existing moderation acts after content reaches recipients, depends on fixed harm taxonomies that cannot be updated without retraining, and rarely covers more than one or two modalities. Organizational settings require the opposite: validation before delivery, team-specific policies that change often, and coverage across every modality a message can arrive in. This paper contributes Convo-Ease, an applied moderation architecture defined by four design decisions and realized as a working system. A pre-delivery gatekeeper validates content structurally in the message-send path, so a message that fails evaluation is never shown to other members. A unified text abstraction reduces every modality to a single representation before evaluation, so one policy layer governs all three. A Policy-as-Prompt mechanism reads organizational rules as plain-language instructions at inference time, so a policy change is a configuration edit rather than a retraining job, and no model training is required at any point. A plugin-based backend separates each moderation task from the model that performs it, so a hosted application programming interface model and a local open-weight model are interchangeable per modality through configuration alone. We validate the central architectural claim on a held-out set of 300 human-labeled messages evaluated live under matched conditions on the same model with no parameter update between them: replacing a generic safety taxonomy with organization-specific natural-language rules raises recall from 46.0% to 76.7% and accuracy from 68.3% to 76.7%, a paired difference that is statistically significant (McNemar's exact $p = 0.004$). The same mechanism holds across fifteen distinct organizational rule sets without any per-rule retraining, and an independent enforcement-intensity control shows a measured, asymmetric effect on the operating point. The quantitative evaluation covers the text policy-evaluation core; the multimodal contribution is architectural and is not accompanied by end-to-end measurement on real image or audio media. Live policy evaluation against a hosted model sustains sub-second batched latency, within the budget required for real-time chat. Convo-Ease demonstrates that an adaptable, multimodal, pre-delivery moderation system can be assembled from interchangeable off-the-shelf models plus an architecture, with no training in the loop, and that this design occupies a niche no existing system fills.

Categories: AI/ML-based decision support systems, Natural Language Processing (NLP)

Keywords: ai moderation, multimodal learning, organizational communication, text-image-audio fusion, rule-based moderation

Introduction

Digital messaging tools sit at the center of how organizations operate. Engineering teams coordinate on Slack or Microsoft Teams, sales teams run on WhatsApp, and customer-facing groups maintain dedicated channels for vendor and client contact. The same flexibility that makes these tools useful is where the operational risk begins. Conversations drift off-task, internal policies are broken, and sensitive information occasionally lands in a channel it should never have

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. Cureus J Comput Sci 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

reached. None of these are exotic failures. They are everyday risks, and the moderation tooling available today does not match them, because most of it was built for public social platforms where the goal is to remove harmful content after a user reports it. In an internal channel, that workflow is already too late: once a confidential document or a policy-violating message is visible, the harm has begun, even if the message is deleted a minute later.

What organizations need is a different shape of moderation. It must screen content before it reaches recipients, apply team-specific policies in real time, adapt as those policies change, and work across whatever modality the content arrives in, whether text, image, or audio, rather than handling only one of them well.

This is a systems problem, not a modeling problem. The individual capabilities required - language understanding, image captioning, and speech transcription - already exist in off-the-shelf models. What does not exist is an architecture that arranges them so an organization gets pre-delivery validation, adaptable natural-language policy, and multimodal coverage at once, without retraining a model each time a rule changes, and without shipping content to an external classifier it does not control. Four limitations recur in current systems. Modality coverage is uneven, with text handled reliably, images partially, and audio rarely. Policy adaptability is poor, since updating a rule typically means retraining or re-engineering the model rather than editing a configuration [1,2]. Deployment context is ignored because methods built for community moderation or agent safety carry assumptions that do not transfer to enterprise chat [3]. And the validation point is wrong: messages are inspected after delivery, so a violating message has already been read by the time the system reacts.

The contribution of this paper is an architecture for real-time multimodal moderation and a working system that realizes it. First, a pre-delivery gatekeeper architecture places moderation structurally in the message-send path as inline middleware, so a message that fails evaluation is persisted only in a withheld state and never enters the feed that other members read; the check-before-publish property is a property of the architecture, not of model accuracy. Second, a Policy-as-Prompt enforcement mechanism reads organizational rules authored in natural language at inference time, with a separate control that adjusts enforcement intensity independently of rule content, so a policy change takes effect on the next message with no retraining anywhere in the system. Third, a plugin-based backend architecture separates each moderation task from the model that performs it, so any modality's model, hosted API or local on-premises, is substituted through a single interface with no change to the moderation logic, which is what lets an organization run identical policy logic against an on-premises model for data-sovereignty reasons. Fourth, a unified text abstraction reduces image and audio to text before evaluation, so a single natural-language policy layer governs all three modalities without separate per-modality moderation logic.

The remainder of this article reviews the related work bearing on the design, presents the architecture and its Policy-as-Prompt and plugin mechanisms, describes the evaluation methodology, reports the architectural properties the system demonstrates and their measured support, and discusses implications, limitations, and positioning against existing systems.

Convo-Ease draws on and departs from several lines of work in automated moderation. The most relevant fall into three themes: text guardrails and policy adaptability, multimodal moderation, and the gap that motivates the present design.

Llama Guard is a safety classifier that scores content against a configurable taxonomy of harm categories, and its zero-shot configurability, expressing a new policy as a taxonomy update rather than a retraining run, is the property that informed our Policy-as-Prompt design [1]. Its limits are equally clear: text only, with no awareness of organizational policy as distinct from generic safety. LlamaFirewall layers prompt-injection detection, behavior auditing, and code inspection for agent safety [2]. The sharpest quantified gap is in policy adaptability: a leading static-taxonomy guardrail drops to 13.1% F1 when asked to enforce free-form, user-defined policies, a direct measure of how far fixed-taxonomy systems sit from what organizational moderation needs [4]. The closest conceptual neighbor to our approach encodes community moderation rules as natural-language instructions and reaches a median accuracy near 64% on Reddit communities, with most residual error attributable to prompt brittleness rather than reasoning [3]. Convo-Ease adopts that conceptual core,

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

ports it to a multimodal enterprise setting, and later reproduces the underlying pattern, a fixed taxonomy underperforming a rule-aware alternative, as a controlled paired comparison within one system, holding the model fixed and varying only the policy content.

Adding images and audio changes the difficulty profile in ways that are not always intuitive. A multimodal benchmark across social-media tasks finds zero-shot performance consistently weak and the modality gap task-dependent rather than uniform [5]. A study comparing video hate detection with image-text meme detection reports a macro-F1 of 0.848 for the video task but an area under the receiver operating characteristic curve of only 0.628 for the meme task under one architecture; it further shows that incorporating transcript-derived text before classification restores video-level performance [6]. Image captioning has been shown to serve the same role, converting a visual moderation problem into a text problem [7], and a training-free defense converts unsafe image inputs to text so a model's existing safety alignment activates, yielding large safety gains with no fine-tuning [8]. These results are the empirical basis in the literature for Convo-Ease's decision to reduce every modality to text before evaluation.

Multimodal guardrails are advancing, but not yet far enough for the enterprise case. A recent extension of a text-only guardrail to image content handles only text and image, operates in English, and processes a single image at a time [9]. Adversarial benchmarks show that jailbreak strategies transfer from text-only to multimodal models with attack success above 50% on specific categories [10], and a multi-domain benchmark of nineteen guardrail models reports wide cross-domain variance and uniform adversarial fragility [11]. Across these lines of work, no existing system combines real-time coverage of text, image, and audio with adaptable natural-language policy enforcement and pre-delivery validation. That unoccupied combination is the niche Convo-Ease targets.

Materials And Methods

System architecture

Convo-Ease is organized around five decisions that together produce the behavior pre-delivery enterprise moderation requires. Modular separation: each modality has its own processing path, and the three paths run independently behind a shared plugin interface, so an installation can adopt text moderation first and add image or audio later without architectural rework. Unified text abstraction: every input is converted to text before it reaches the policy evaluator, so text arrives as text, images are reduced to a short plain-text description, audio is transcribed and summarized, and a single policy layer governs all three. Externalized policy in two dimensions: rules are kept outside the model, authored in natural language by a group administrator, stored with the group record, and assembled into the evaluation prompt at inference time; a separate sensitivity setting (Strict, Moderate, or Relaxed) controls how aggressively the same rules are enforced at the margin. Rules and sensitivity are edited independently, and their effects are measured separately in the Results and Discussion. Structural gatekeeping: the check-before-publish property is enforced in the message-send path itself, so a message is evaluated before it is persisted in a deliverable state, and a flagged message is never returned in the visible feed. Built-in explainability: every decision carries a structured reason, so a blocked message records the rule-grounded explanation that triggered it, which makes the system auditable and gives an affected member a basis for appeal.

Convo-Ease moves a message through a sequence of stages, shown in Figure 7. At input, the interface receives a text, image, or audio submission for a given group. In preprocessing and modality reduction, text passes through deterministic pre-filters (described in the section "The Policy-as-Prompt Mechanism"), an image is sent to a vision-language backend that returns a one- or two-sentence plain-text description, and audio is transcribed and then summarized, so after this stage, every modality is represented as text. In policy integration, the unified text is assembled into an evaluation prompt together with the group's rules, the configured sensitivity, a context-use mode, and a bounded slice of recent conversation. In core moderation, the prompt is submitted to the language-model evaluator, which returns a verdict parsed into an allow or block decision with a reason. In output and enforcement, an allowed message is

How to cite this article:

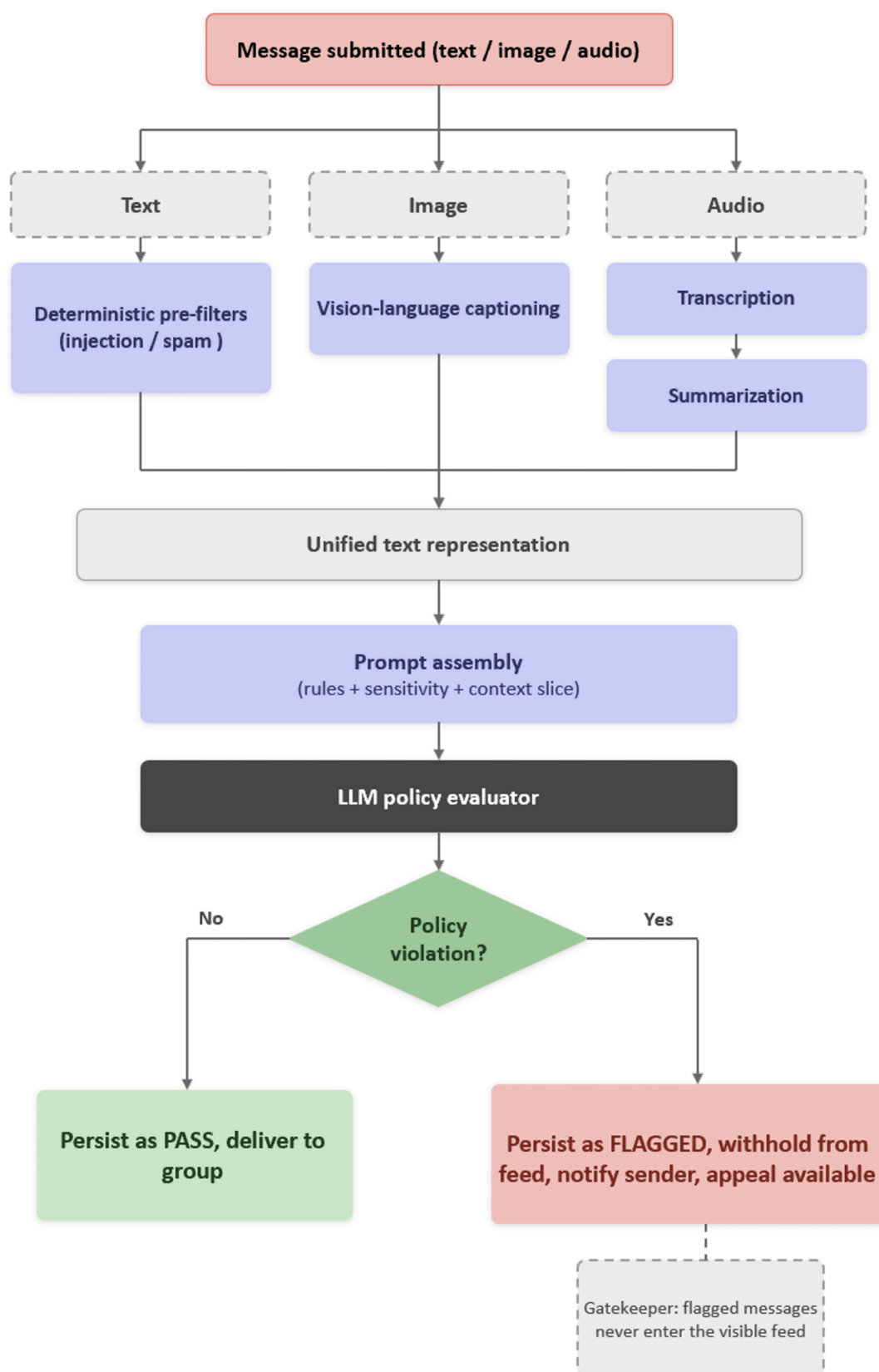
Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

persisted and delivered, while a blocked message is persisted with a reason, withheld from the feed, and the sender is notified. Administrators author and edit rules, set sensitivity, and inspect a dashboard aggregating pass and flag rates, reasons, and per-member indicators from the decision log.

The architecture assigns each modality a role rather than a fixed model; this is summarized in Table 7. The policy evaluator must reliably follow natural-language instructions over long contexts, since it reads the full policy, a conversation slice, and the message in a single pass. The image captioner must produce a faithful short description, and the transcriber must have an error rate and language coverage adequate for enterprise use without domain fine-tuning [12,13]. Vision-language pre-training of the BLIP-2 family bridges image and text at real-time latency [14], and the fusion taxonomy of the multimodal representation literature maps onto these role assignments [15]. The plugin architecture described below formalizes the interface that makes these roles model-agnostic.

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>



How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. Cureus J Comput Sci 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

FIGURE 1: Convo-Ease moderation pipeline

A submitted message is reduced to text (images captioned, audio transcribed, and summarized), evaluated against the assembled prompt by the language-model policy evaluator, and either delivered or withheld.

PASS: the message satisfies the group's policy and is persisted and delivered to the group feed normally.

FLAGGED: the message violates the policy and is blocked before delivery, persisted in a withheld state together with the rule-grounded reason that triggered the block, and made available to the sender and to administrators for audit and appeal. It never enters the feed that other members read.

LLM, Large Language Model

Task	Prototype binding	Execution	On-premises alternative
Policy evaluation	Large open model via hosted API	Hosted	Local open-weight LLM
Image captioning	Vision-language model via hosted API	Hosted	Local BLIP-2 / LLaVA-class
Audio transcription	Whisper	Local	Same
Policy store	NL rules + sensitivity per group	Local	Same
Decision parsing	Structured PASS/FLAGGED, fail-safe	Local	Same
Pre-filters	Deterministic injection/spam/leetspeak	Local	Same
Application layer	REST send-path gatekeeper	Local	Same

TABLE 1: Convo-Ease task-to-model bindings are shown in the prototype. The architecture is backend-agnostic; on-premises open-weight equivalents use the same interfaces

API, Application Programming Interface; BLIP-2, Bootstrapping Language-Image Pre-training 2; LLaVA, Large Language and Vision Assistant; LLM, Large Language Model

The Policy-as-Prompt mechanism

The Policy-as-Prompt mechanism turns an organizational rule into a moderation decision without retraining. It rests on the in-context instruction-following ability of modern language models [16] and is engineered to control the prompt brittleness that limits the natural-language policy approach in prior work [3].

A policy is a block of natural-language rules attached to a group, authored and edited by the administrator. Each group also carries a sensitivity setting, Strict, Moderate, or Relaxed, which is a separate control from the rules. The rules define the content boundary, what is or is not allowed. The sensitivity defines the enforcement intensity at that boundary, how aggressively borderline content is treated once a rule could plausibly apply. Because the policy lives with the group record rather than in model weights, changing a rule or the sensitivity takes effect on the next message with no retraining.

How to cite this article:

At inference time, the system assembles one evaluation prompt from the rules, the sensitivity instruction, a context-use mode, the recent-conversation slice, and the message in unified-text form. The prompt instructs the model to treat the rules as the primary decision boundary, to use context only for resolving short replies, pronouns, and references, and not to flag a message merely for changing the subtopic. It includes explicit handling for emoji, evaluated by intent rather than appearance, and instructs the model to judge non-English content and character substitutions by meaning rather than literal characters. At inference time, the evaluation prompt consists of five components: (1) administrator-defined rules, (2) moderation sensitivity, (3) context-use mode, (4) recent conversation context, and (5) the new message. It instructs the model to treat the administrator rules as the primary decision boundary, use conversation context only to resolve references or ambiguity, evaluate emoji and non-English content by semantic meaning, and return only one of two structured outputs: PASS or FLAGGED <reason>.

The sensitivity instruction resolves to one of three texts: Strict flags borderline content and leans toward blocking when uncertain, Moderate (the default) enforces consistently without over-flagging, and Relaxed flags only clear and meaningful violations. The context-use mode is set to required for short or referential messages, and reference-only otherwise, so the evaluator leans on context only when the message cannot be judged alone.

The reply is parsed into a verdict. A reply beginning PASS is an allow; a reply beginning FLAGGED is a block whose trailing text becomes the stored reason. An empty or malformed reply is treated as a block with an explanatory reason rather than a silent pass. This is deliberate: in a pre-delivery gatekeeper, an ambiguous model output must not result in unmoderated delivery. The system also fails closed at the transport level, so a backend error or timeout blocks the message rather than delivering it unchecked.

Before any model call, a text message passes three deterministic filters. An injection filter matches known manipulation patterns (instructions to ignore prior directions, assume a new role, or force a PASS) and blocks with an injection-specific reason. A spam filter catches placeholder text and degenerate repetition. A leetspeak normalizer rewrites common substitutions and passes the decoded form to the model as a hint. These provide a fast, model-independent first pass whose measured standalone coverage is reported in the Results and Discussion; they are a supplement to, not a replacement for, the model layer.

Algorithm 1 formalizes the per-message decision for the text path. The image and audio paths are identical from the unified-text stage onward, differing only in the modality-reduction step that produces that text.

Algorithm 1: Pre-delivery moderation decision (text path)

Input: message m , group rules R , sensitivity s , recent context C

Output: verdict v in {PASS, FLAGGED(reason)}

```

1: if  $R$  is empty then return PASS # no policy, no gate
2: if injection_pattern( $m$ ) then return FLAGGED(injection_reason)
3: if spam_pattern( $m$ ) then return FLAGGED(spam_reason)
4:  $m\_dec := \text{normalize\_leetspeak}(m)$  # decoded hint if changed
5: mode := context_mode( $m$ ) # required | reference_only
6: prompt := assemble( $R$ ,  $s$ , mode,  $C$ ,  $m$ ,  $m\_dec$ )
7: out := LLM_evaluate(prompt) # constrained: PASS | FLAGGED <reason>
8:  $v := \text{parse}(out)$  # fail-safe on empty/ambiguous
9: return  $v$ 

```

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

Plugin-based backend architecture

A processing plugin defines a moderation task, moderates text, describes an image, and transcribes audio, independent of any model that performs it. A backend factory binds each task to a concrete model at configuration time, either a hosted inference API or a local on-premises model, selected per modality. This separation is the architectural property, and it holds regardless of which model is plugged in at a given time. It directly answers a limitation shared by the text-only and multimodal systems reviewed above: a moderation architecture tied to one model's release cycle.

The same plugin interface accepts a hosted model behind an API or a local open-weight model, chosen per modality through configuration. Also, the plugin interface supports either hosted inference APIs or local open-weight models, selected independently for each modality. Changing the backend requires only a configuration update and does not alter the moderation logic. This is the same edit-not-retrain property that Policy-as-Prompt gives for rules, extended to models.

Commercial moderation services require sending content to an external server for evaluation, which conflicts with data-sovereignty requirements in regulated industries, discussed further in the Results and Discussion. The plugin separation means an organization can run identical policy logic against an on-premises model with no architectural change. The prototype demonstrates both modes: it runs its text evaluator and image captioner through a hosted inference API and its transcriber locally, and the same interfaces accept fully local models for a sovereign deployment. The interchangeability is exercised in this paper rather than asserted: the evaluation was produced by running the live-hosted configuration against a real model over real network round-trip times, with wall-clock latency captured per evaluation batch. Figure 2 shows the plugin and backend structure.

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

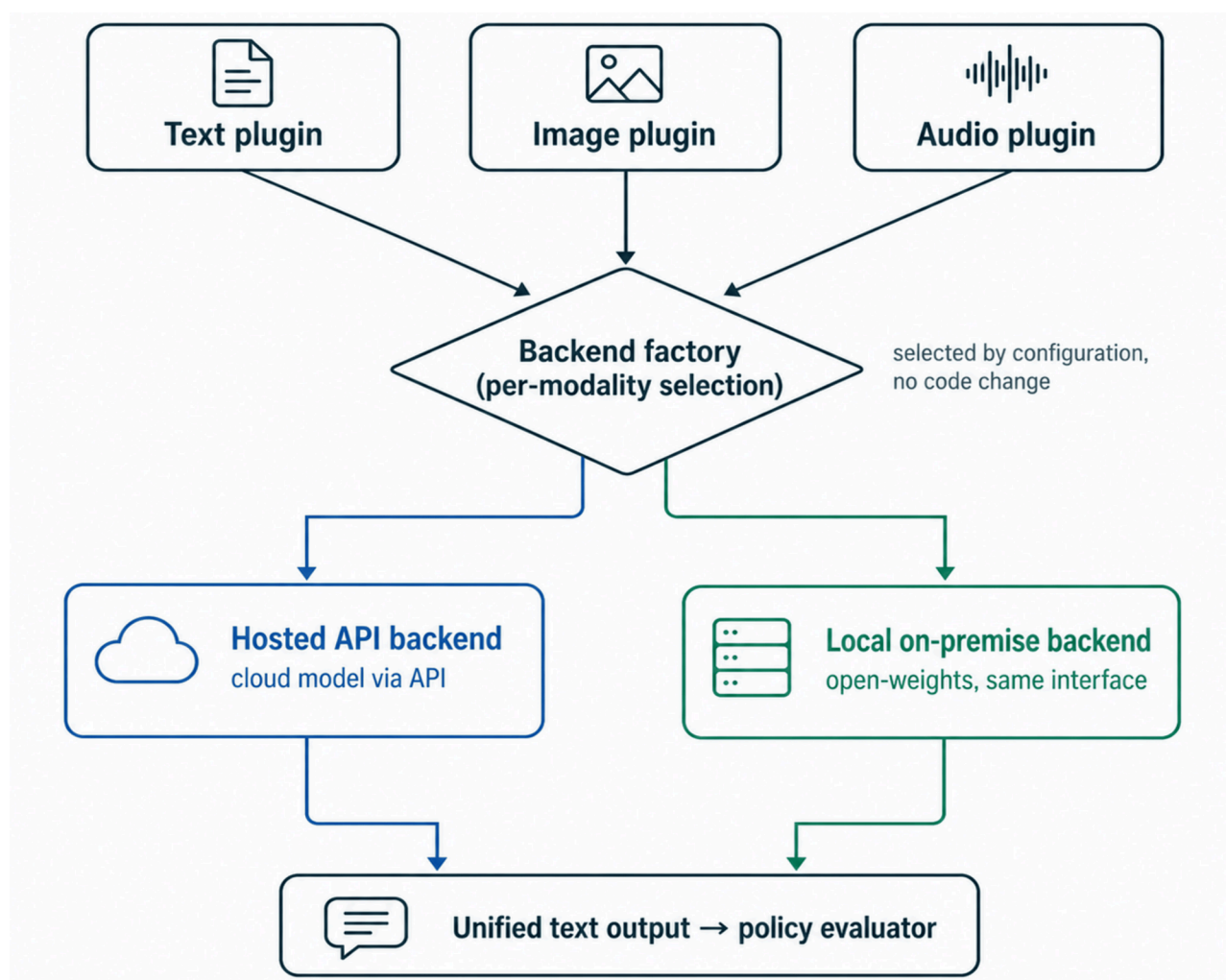


FIGURE 2: Backend-agnostic plugin architecture

Each modality plugin binds to either a hosted API backend or a local on-premises backend through the backend factory, selected by configuration with no change to the moderation logic.

Unified text output: the single plain-text representation that every modality is reduced to before policy evaluation. Text messages pass through unchanged, images are converted by a vision-language model into a short textual description, and audio is transcribed and then summarized, so that one policy evaluator reads all three modalities in the same form.

API, Application Programming Interface

Evaluation methodology

The evaluation is designed to test the architecture's claims, not to rank a model. It is organized so the central claim, that reading organizational policy as a prompt changes moderation outcomes, is isolated as a controlled variable, and so the system can be assessed without exposing real organizational chat.

The evaluation grounds itself in public, human-labeled moderation corpora mapped into the policy-conditioned format the system consumes. A conversion pipeline ingests the Jigsaw Toxic Comment Classification data [17] and the Stormfront-derived HateSpeech18 sentence-level hate-speech corpus [18] and emits records of the form (source, message, rules, context, label). Each message is paired with one of fifteen realistic organizational rule sets spanning respect and anti-harassment, anti-discrimination, anti-spam, on-topic, English-only, and composite multi-rule policies, so a labeled message is evaluated under a concrete named policy rather than an abstract taxonomy. The pairing is

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. Cureus J Comput Sci 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

randomized. Each message is assigned a rule set by uniform random draw from the fifteen, with no content-based matching between the message and the rule it is evaluated against. Ground-truth labels are carried directly from the source corpora's human annotations, thresholded to a binary verdict, so a message annotated as toxic or hateful in the source data carries a violating label and all others carry an allowable label. The label therefore expresses the source annotation rather than a policy-specific judgment authored for the assigned rule set, and the benchmark should be read accordingly. Because the pairing and the labels are held identical across all four evaluation conditions, any mismatch between a label and the rule it is paired with is present in equal measure in every condition and cannot account for a difference between them. This is what makes the paired comparison interpretable despite the proxy construction, and the mismatch is treated as a limitation of the benchmark rather than a property of the architecture. Human annotation is preserved as ground truth, and no private conversational data is used. The held-out evaluation set contains 300 messages, balanced at 150 allowable and 150 violating. The benchmark is derived from the publicly available Jigsaw Toxic Comment Classification and HateSpeech18 datasets. Raw message text is not redistributed due to the redistribution terms of the original datasets.

Because the source corpora's redistribution terms prevent reproducing original message text, the following synthetic examples illustrate the categories of content the benchmark contains. They are constructed for illustration and are not drawn from the datasets. A message annotated as violating in the source data resembles a direct personal insult aimed at another participant, such as telling a member they are worthless and should leave the group, or a statement asserting that people of a particular nationality do not belong in the community. A message annotated as allowable resembles an ordinary workplace exchange, such as asking whether anyone has resolved a dependency conflict in a build or a routine request to move a meeting to a later slot. Under the randomized pairing described above, either type of message may be evaluated against any of the 15 rule sets, so a benign scheduling message may be assessed under an anti-discrimination policy and an insult may be assessed under an English-only policy.

To isolate what a change in policy content contributes, independent of the model, the same live model evaluates the identical 300 messages under four conditions: a generic safety taxonomy and organizational rules at three sensitivity settings (Strict, Moderate, Relaxed), with no parameter update between conditions. Because all four score the same items, this is a paired design, which permits McNemar's test to assess the significance of any difference in accuracy between two conditions, rather than an informal comparison of independent accuracies.

Evaluation was run live against the deployed text-moderation plugin, which calls a hosted large open model (OpenAI gpt-oss-120b served via the NVIDIA NIM API) [19] with a temperature of 0. Greedy decoding at a temperature of 0 makes the model near-deterministic. Although no effective random seed could be specified for the hosted model endpoint, the evaluation was executed under a fixed inference configuration, and a machine-recorded completion timestamp was retained for reproducibility. Reported figures are recomputed directly from the 1,200 stored per-message predictions (300 messages $\times$ four conditions). Significance tests use McNemar's paired test, exact binomial when the smaller discordant count is below 25, and the continuity-corrected chi-square otherwise, with 95% confidence intervals from a 10,000-sample bootstrap. The raw predictions, evaluation scripts, and supporting code used to reproduce the reported metrics are publicly archived in the Convo-Ease Zenodo repository [20], so every reported result can be independently recomputed.

For reproducibility, the live benchmark evaluation was completed on June 23, 2026, with a machine-recorded completion timestamp of 2026-06-23T13:44:28.124278 (+05:30). All experiments used the hosted openai/gpt-oss-120b model through the NVIDIA NIM OpenAI-compatible API. A separate model version identifier was not exposed for this hosted endpoint. The inference configuration remained fixed throughout the evaluation (temperature = 0, top_p = 1), and the same prompt template and policy configuration were applied across all benchmark conditions. All evaluations used in-context inference only. No model training, weight updates, or parameter fine-tuning were performed. Although the OpenAI-compatible request schema accepts a seed parameter, deterministic seeding is not implemented for the hosted model endpoint used in this study. Consequently, no effective random seed could be specified. The evaluation was performed using Convo-Ease version 4.2 (source revision 49467828986aede13ae2976ae5a16dd1350eb24a).

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

This study measures four things directly: the effect of policy content on decisions, the consistency of that mechanism across rule sets, the effect of the sensitivity control, and live policy-evaluation latency; it also reports the standalone coverage of the deterministic pre-filter layer. End-to-end evaluation on real image and audio media is outside the scope of this study. The multimodal contribution is architectural and is supported by the system design and prior work on text-reduction moderation [6-8].

The benchmark harness submits held-out items to the same hosted model under each of the four policy conditions in fixed groups of ten, with up to three groups dispatched concurrently, so the latency reported in Table 6 is wall-clock round-trip time per group of ten rather than per single message. No retried requests were recorded across the full run.

Implementation details

This section records the implementation the architectural claims rest on, so that the structural properties asserted in the Results and Discussion can be checked against the archived source rather than taken on the basis of description [20].

Plugin and backend interfaces: a processing plugin is defined by an abstract base class exposing three members: an input schema accessor, an output schema accessor, and an abstract process method taking the input and an optional context. Three concrete plugins exist, one per modality, for text moderation, image moderation, and audio transcription. A plugin does not construct or import a model. It resolves one at initialization through a backend factory, which reads a model configuration and returns a backend satisfying a common generation interface. That single indirection is the whole of the substitution mechanism: changing which model performs a task is a change to the configuration passed to the factory, and no moderation logic is touched. All three plugins resolve their backends through the same factory, which is why a modality can be bound to a hosted API or to a local model independently of the other two.

Send-path integration and the withheld state: the message-send endpoint executes a fixed order of operations. It reads the sender, message, and group from the request, retrieves the group record including its rules and sensitivity, retrieves the recent visible messages of that group as context, resolves the text moderation plugin, and, if the group has rules, invokes moderation before any persistence of a deliverable message occurs. A message that passes is stored with status PASS. A message that is flagged is stored with status FLAGGED together with the reason the evaluator returned. The group feed is rendered from a visibility query that admits only the PASS and DELETED statuses, so a FLAGGED record is excluded by the query that builds the feed rather than by a downstream presentation rule. This is the code-level basis for the structural gatekeeping claim: the withheld state is a stored value the visibility path does not select, not a flag consulted at render time. If the moderation backend raises an error, the endpoint returns an unavailable response, and the message is not delivered, so a backend failure blocks rather than passes.

Context slice and context-use mode: the context passed to the evaluator is the six most recent visible messages of the group. The context-use mode is selected by a deterministic test on the incoming message rather than by the model. Context is marked required when the message is four tokens or fewer, when it matches a short reference phrase, when it is eight tokens or fewer and contains a pronoun marker, or when it is a question of six tokens or fewer. Otherwise, the mode is reference-only. The intent is that the evaluator leans on conversation history only for messages that cannot be judged standing alone.

Deterministic pre-filters: the injection filter matches against 28 substring patterns covering instructions to disregard prior directions, to assume a different role, or to force a particular verdict. The spam filter combines four placeholder-text patterns with two degeneracy heuristics, one on distinct characters for messages of at least eight characters and one on distinct words for messages of at least five words. The leetspeak normalizer applies a 13-entry character substitution map and passes the decoded form to the evaluator as a hint rather than replacing the original message. All three run before any model call and are model-independent, which is what makes their cost negligible and their coverage limited.

Inference configuration and stack: the evaluation used the hosted gpt-oss-120b model through the NVIDIA NIM OpenAI-compatible API at temperature 0 and top_p 1 [19]. The prototype application layer is a Python 3.10 Flask service with a file-backed message and group store, and the audio path invokes a local Whisper base model through its Python API on a

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

temporary decoded audio file [12]. None of the reported properties depend on the storage layer, which is a prototype convenience rather than a design element.

Results And Discussion

Experimental setup

All quantitative results reported in this section were obtained from the live benchmark evaluation described in the Materials and Methods. The evaluation used the hosted openai/gpt-oss-120b model through the NVIDIA NIM API under fixed zero-shot inference settings with no model training or fine-tuning. Performance was measured on a held-out benchmark of 300 messages derived from the Jigsaw Toxic Comment Classification and HateSpeech18 datasets, evaluated under four policy conditions (generic taxonomy and organizational rules with Strict, Moderate, and Relaxed sensitivity), yielding 1,200 stored prediction outcomes. Unless otherwise stated, reported metrics are computed directly from these benchmark predictions.

The gatekeeper enforces structurally

A flagged message is excluded from the visibility query that renders the group feed and is never returned to any member other than the sender and administrators. This holds independent of model accuracy: it is a fact about where moderation sits in the send path, verified at the code level, not a statistical outcome that could degrade on a harder input distribution. It is reported first because every result that follows concerns how well the gate decides, not whether the gate holds once it has decided.

Policy content is a controllable variable with a significant effect

Table 2 reports the four metrics on the same 300 messages, scored by the same live model, under a generic safety taxonomy versus the organization's own rules at moderate sensitivity. The effect is concentrated in recall. Reading organizational rules instead of a generic taxonomy raises recall from 46.0% to 76.7%, a gain of 30.7 percentage points, while precision falls more modestly from 83.1% to 76.7%. Net accuracy rises from 68.3% to 76.7%. The Matthews correlation coefficient, which summarizes the full confusion matrix in a single value, rises from 0.410 to 0.533 over the same comparison. The underlying counts are 69 true positives, 14 false positives, 136 true negatives, and 81 false negatives under the generic taxonomy, against 115, 35, 115, and 35, respectively, under organizational rules. Balanced accuracy is not reported separately because the held-out set is balanced at 150 allowable and 150 violating, on which balanced accuracy is identical to accuracy by construction. On the 300 paired predictions, the accuracy difference is significant (McNemar's exact $p = 0.004$, 95% bootstrap CI [+3.0, +14.0] percentage points; discordant pairs: 48 in favor of the organizational rules against 23 in favor of the generic taxonomy), as reported in Table 5. The practical reading is that a generic safety taxonomy misses more than half of what an organization's own rules would catch: business-language violations, off-topic drift, and English-only breaches, none of which map onto a public-platform harm category. Policy-as-Prompt closes most of that gap with a change of prompt content alone, with no model training, fine-tuning, or parameter updates. This is the controlled, single-system version of the pattern established at field scale by the observation that fixed-taxonomy guardrails collapse under free-form policy [4]; here, the model is held constant, and policy content is the only variable that moves. Figure 3 shows the tradeoff and its confidence interval.

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

Condition	Accuracy	Precision	Recall	F1	MCC
Generic taxonomy	68.3%	83.1%	46.0%	59.2%	0.410
Organizational rules (moderate)	76.7%	76.7%	76.7%	76.7%	0.533
Delta	+8.3 pp	-6.4 pp	+30.7 pp	+17.4 pp	+0.123

TABLE 2: Moderation performance on the identical 300 held-out messages, using the same live model, under a generic safety taxonomy versus organizational rules at moderate sensitivity. Recomputed from stored per-message predictions

Precision is the proportion of messages flagged by the system that are genuine violations. Recall is the proportion of genuine violations that are flagged by the system. F1 is the harmonic mean of precision and recall. MCC summarizes the full confusion matrix on a scale from -1 to $+1$, with 0 indicating chance-level agreement. MCC, Matthews Correlation Coefficient; pp, Percentage Points

All figures are on the same 300 held-out messages, scored by the same model with no parameter change between conditions.

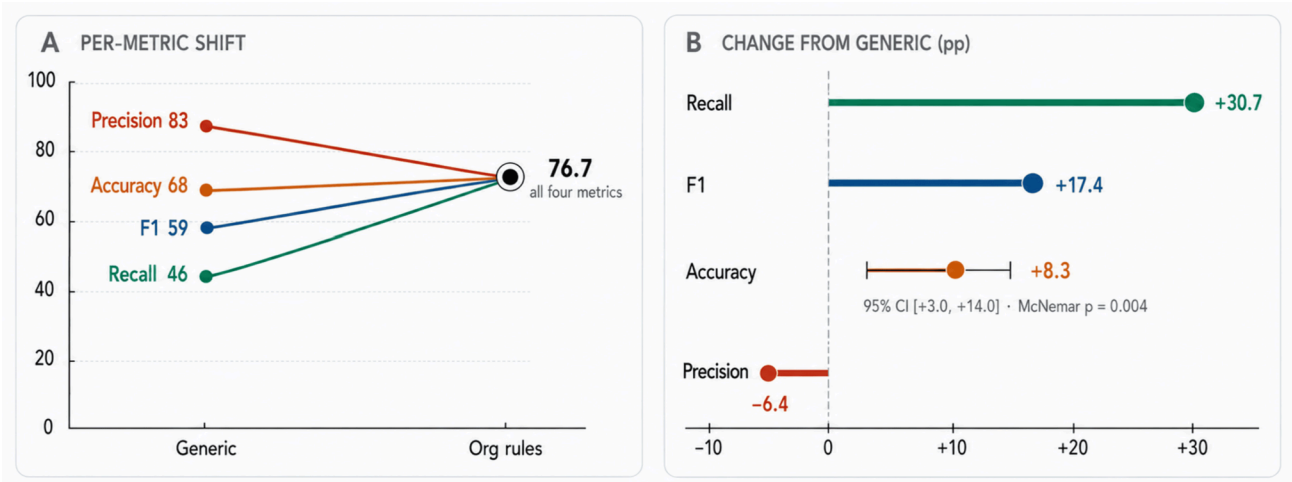


FIGURE 3: Reading organizational rules rebalances the moderator

Policy-content isolation on the same 300 held-out messages, using the same model with no retraining. Panel A: each metric shifts from the generic taxonomy to the organizational rules, with all four converging at 76.7%. Panel B: change from the generic baseline in percentage points; the accuracy interval and significance are shown. Statistical significance was assessed using McNemar’s paired test on the 300 matched predictions. CI, Confidence Interval; pp, Percentage Points

The mechanism generalizes across policies without retraining

The same mechanism was evaluated under all fifteen distinct organizational rule sets in the held-out data, with no per-rule tuning and no model change. Table 3 reports, for each rule set, accuracy under organizational rules with 95% Wilson confidence intervals, the accuracy the generic safety taxonomy achieved on the identical messages, and the paired

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. Cureus J Comput Sci 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

discordant counts between the two conditions.

Two questions bear on generalizability, and the table answers them separately. The first is whether one mechanism operates at a consistent rate across policies it was never tuned for. Pooled across all fifteen rule sets, the mechanism classifies 230 of 300 messages correctly, which is the 76.7% reported in Table 2; the per-rule figures partition those same predictions rather than forming an independent measurement. Per-rule accuracy ranges from 56.5% to 93.8%, but the per-rule samples are small, between 15 and 32 messages, and that spread is not larger than sampling variation alone would produce. A chi-square test of homogeneity across the fifteen per-rule outcome tables does not reject a common underlying accuracy (chi-square = 20.58, df = 14, p = 0.11). Because twelve of the thirty expected cell counts fall below five, a 20,000-sample permutation test of the same statistic was also run and gives p = 0.11. The evidence is consistent with a single mechanism operating at one rate across all fifteen policies, and the ordering in Table 3 should not be read as a ranking.

The second question is whether the benefit of reading organizational rules is carried by a few rule sets or holds broadly. Because both conditions scored the identical messages, this is answerable within each rule set. Organizational rules are at least as accurate as the generic taxonomy in fourteen of the fifteen rule sets: accuracy improves in eight, is unchanged in six, and falls in one. Across the nine rule sets where the two conditions differ at all, the improvement is significant on a sign test (8 of 9, p = 0.039, with the six tied rule sets excluded). The median per-rule change is +3.1 percentage points, and the range runs from -12.5 to +31.3 points. The single rule set where accuracy falls is the study group policy, where two messages of sixteen change from correct to incorrect. The per-rule discordant counts sum to 48 and 23, reproducing the aggregate paired comparison in Table 5, so the per-rule view decomposes that result rather than restating it.

The per-rule p-values in Table 3 are reported for completeness and are individually underpowered and uncorrected for multiple comparisons; no rule set should be read as significant on its own. The claim the table supports is at the level of the set: the mechanism runs across all 15 policies without any per-rule modification, at a rate statistically indistinguishable across them, and the advantage of organizational rules over a generic taxonomy is distributed across policies rather than concentrated in a few. Because rule sets were assigned to messages at random rather than matched to message content, per-rule differences also reflect which messages happened to fall under each rule, and no causal attribution to rule complexity is claimed.

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

Rule set (abbreviated)	n	Accuracy, org rules (95% CI)	F1 (flagged)	Accuracy, generic taxonomy	Δ (pp)	Discordant (b/c)	p
Respectful, no insults	26	92.3% (75.9% to 97.9%)	93.3%	69.2%	+23.1	6/0	0.031
No explicit/violent content	16	93.8% (71.7% to 98.9%)	93.3%	62.5%	+31.3	5/0	0.062
Respect, no harassment/bullying	17	88.2% (65.7% to 96.7%)	87.5%	88.2%	0.0	1/1	1.00
Community: no hate/threats	20	85.0% (64.0% to 94.8%)	87.0%	55.0%	+30.0	6/0	0.031
English only	15	80.0% (54.8% to 93.0%)	84.2%	80.0%	0.0	2/2	1.00
Group guidelines: respect, no spam	21	81.0% (60.0% to 92.3%)	77.8%	57.1%	+23.8	6/1	0.125
No spam/unsolicited promotion	17	82.4% (59.0% to 93.8%)	76.9%	82.4%	0.0	1/1	1.00
No discrimination	19	78.9% (56.7% to 91.5%)	75.0%	73.7%	+5.3	3/2	1.00
Stay on topic	21	76.2% (54.9% to 89.4%)	76.2%	66.7%	+9.5	4/2	0.688
Business: professional language	32	75.0% (57.9% to 86.7%)	75.0%	71.9%	+3.1	4/3	1.00
Study group: on-topic, no aggression	16	75.0% (50.5% to 89.8%)	75.0%	87.5%	-12.5	0/2	0.500
No hate speech/personal	19	68.4% (46.0% to 84.6%)	72.7%	63.2%	+5.3	2/1	1.00

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. Cureus J Comput Sci 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

nal attacks							
No self-promotion/personal links	19	63.2% (41.0% to 80.9%)	58.8%	63.2%	0.0	3/3	1.00
Programming group: multi-clause	19	57.9% (36.3% to 76.9%)	55.6%	57.9%	0.0	2/2	1.00
Composite: insults+promotion+relevance	23	56.5% (36.8% to 74.4%)	58.3%	56.5%	0.0	3/3	1.00
All rule sets pooled	300	76.7%	76.7%	68.3%	+8.3	48/23	0.004

TABLE 3: Cross-policy consistency and per-rule paired comparison. Per-rule-set accuracy under organizational rules at moderate sensitivity across all 15 rule sets, using one shared mechanism with no per-rule retraining, alongside the accuracy achieved by the generic safety taxonomy on the identical messages within each rule set

n: number of messages in the held-out set for that rule set.

CI: 95% Wilson confidence interval for accuracy. Wider intervals reflect the relatively small sample sizes of some rule categories.

pp: percentage points. Δ is the accuracy of the organizational rules condition minus that of the generic taxonomy condition on the same messages, computed from counts.

b/c: the two discordant-pair counts in McNemar's test, where b is the number of messages the organizational rules condition classified correctly, and the generic taxonomy did not, and c is the reverse. The columns sum to 48 and 23 across all rule sets, reproducing the aggregate comparison in Table 5.

p: McNemar exact binomial p value within that rule set. These per-rule tests are individually underpowered and are not corrected for multiple comparisons; they are reported for completeness, and no single rule set should be read as significant on its own.

The pooled row is the aggregate over all 300 held-out messages and matches Table 2; no confidence interval is given for it because it is the full sample rather than a subgroup

Enforcement intensity is a second independent control

Table 4 reports the four metrics across the three sensitivity settings with the organizational rules held fixed. Strict and Moderate are statistically indistinguishable (accuracy 77.0% versus 76.7%; McNemar's $p = 1.00$). Relaxed differs substantially and significantly: accuracy falls to 65.7%, and recall drops from 76.7% to 35.3%, while precision rises to 89.8% (McNemar's continuity-corrected $p = 0.0008$, 95% bootstrap CI $[-17.3, -5.0]$ percentage points against Moderate), as reported in Table 5. The finding is specific. Sensitivity is not a symmetric dial where Strict and Relaxed move equal and opposite amounts from Moderate. It behaves as a one-sided loosening control: tightening from Moderate to Strict buys essentially nothing on this data, while loosening to Relaxed gives up more than a third of the system's recall for a gain in precision. The actionable guidance for an administrator is to treat Moderate as the effective default and reserve Relaxed for channels that deliberately prioritize low-friction, high-precision enforcement over catching every violation. That this control is exposed at all, and that its effect is measurable and interpretable, is the architectural point. Figure 4 shows the sweep and its significance.

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. Cureus J Comput Sci 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

Sensitivity	Accuracy	Precision	Recall	F1	MCC
Strict	77.0%	76.8%	77.3%	77.1%	0.540
Moderate	76.7%	76.7%	76.7%	76.7%	0.533
Relaxed	65.7%	89.8%	35.3%	50.7%	0.394

TABLE 4: Enforcement intensity sweep: the same 300 held-out messages were evaluated under fixed organizational rules, with only the sensitivity setting varied

Precision: the proportion of messages flagged by the system that were genuine violations.
Recall: the proportion of genuine violations that were flagged by the system.
F1: the harmonic mean of precision and recall.
MCC, Matthews Correlation Coefficient

Metrics are defined as in Table 2. The organizational rules are held fixed, with only the sensitivity setting varied.

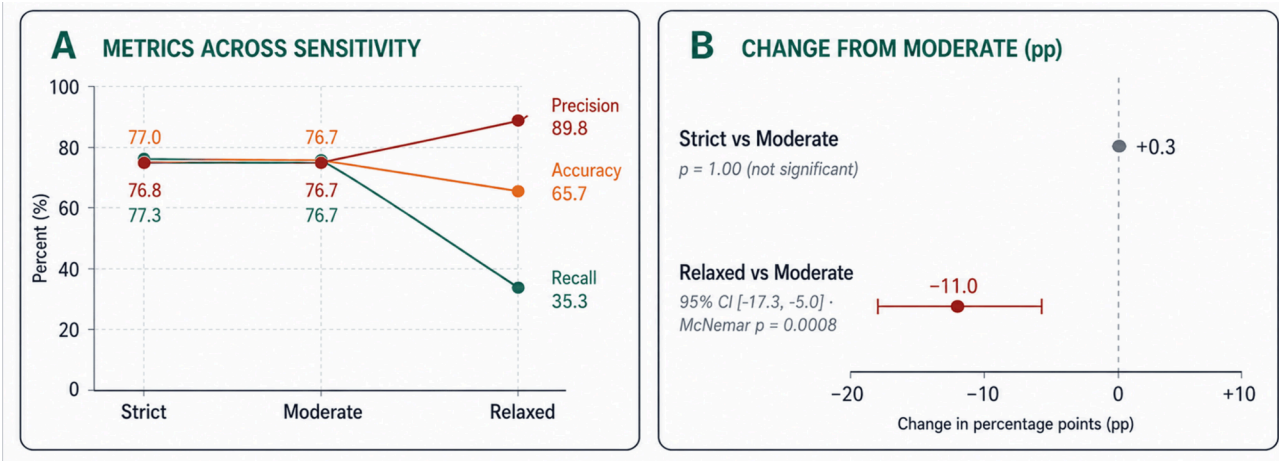


FIGURE 4: Sensitivity sweep (hybrid panel)

Enforcement intensity sweep with organizational rules held fixed.
Panel A: precision, recall, and accuracy across Strict, Moderate, and Relaxed settings.
Panel B: change from Moderate in percentage points, with McNemar's significance. Strict versus Moderate is not significant ($p = 1.00$); Relaxed versus Moderate is significant ($p = 0.0008$).
CI, Confidence Interval; pp, Percentage Points

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. Cureus J Comput Sci 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

Comparison	Δ Accuracy	Discordant (b/c)	p-value	95% CI
Organizational rules vs generic taxonomy	+8.3 pp	48/23	0.004 (exact)	[+3.0, +14.0] pp
Strict vs Moderate	+0.3 pp	1/0	1.00 (exact)	[+0.0, +1.0] pp
Relaxed vs Moderate	-11.0 pp	29/62	0.0008 (χ^2)	[-17.3, -5.0] pp

TABLE 5: Paired McNemar's significance tests on the 300 matched predictions in each comparison. Exact binomial test when the smaller discordant count is below 25; continuity-corrected chi-square test otherwise. CIs derived from a 10,000-sample bootstrap

b/c: the two discordant-pair counts in McNemar's test, where b is the number of messages the organizational rules condition classified correctly, and the generic taxonomy did not, and c is the reverse.
McNemar's test compares two conditions on the same items.
CI, Confidence Interval; pp, Percentage Points

McNemar's test compares two conditions on the same items; a low p-value indicates that the accuracy difference is unlikely to be due to chance. The exact binomial test is used when the smaller discordant count is below 25, and the continuity-corrected chi-square test is used otherwise.

The unified text abstraction holds across modalities

A text-bypass check, in which image and audio inputs are reduced to text and then routed through the same text-moderation path, demonstrates a specific and narrower claim than end-to-end multimodal accuracy: that the downstream policy-evaluation step performs consistently once a faithful text reduction exists, regardless of the modality from which that reduction originated. It does not demonstrate that the upstream captioning and transcription models themselves produce faithful reductions of real images or real audio; that is a separate, not-yet-measured claim. The literature on text reduction as a moderation strategy [6-8] is cited as precedent for expecting the full pipeline to recover comparable performance, not as a substitute for measuring it directly.

The plugin architecture runs live in real time

The evaluation was produced by the live-hosted configuration, and the policy-evaluation latency was captured per batch as a real API round-trip time. Table 6 reports the distribution. Across the four conditions, batched policy evaluation averaged between 436 and 484 ms per ten-message batch, with maxima under 650 ms, against the hosted model. Sub-second batched evaluation at the core of the pipeline demonstrates that the policy evaluator, to which every modality funnels after text reduction, sustains the throughput required for real-time chat. Because the latency reported here is genuine wall-clock timing of live calls rather than an estimate, it is also the concrete evidence behind the plugin architecture's interchangeability claim: this is what the hosted configuration looked like when actually run. Figure 5 shows the per-condition latency against the real-time budget.

How to cite this article:

Condition	Mean (ms)	Median (ms)	Min (ms)	Max (ms)
Generic taxonomy	436	435	302	584
Organizational (moderate)	470	470	341	591
Organizational (strict)	484	485	349	647
Organizational (relaxed)	455	453	302	648

TABLE 6: Live policy-evaluation latency against the hosted model, per 10-message batch, by condition. Measured as genuine wall-clock round-trip time in the live benchmark

Latency is the wall-clock time for one round trip to the hosted model per 10-message batch, not per single message. All conditions were run on the same held-out set.

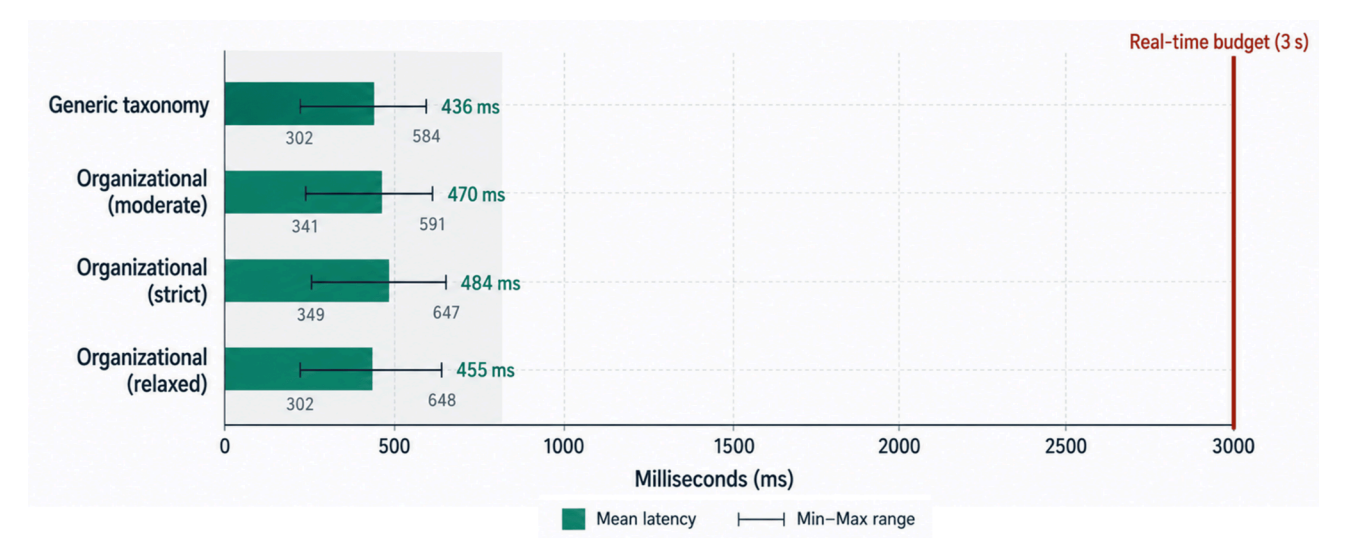


FIGURE 5: Live policy-evaluation latency

Live policy-evaluation latency per 10-message batch, measured as the wall-clock time for one round trip to the hosted model. Bars show mean latency; whiskers show the minimum-to-maximum range. The vertical line marks the three-second real-time budget.

The deterministic and model layers combine to a defensible robustness floor

The deterministic pre-filter is a fast, model-independent first pass, positioned as a supplement to the model layer rather than as the primary defense. Measured against a hand-authored set of ten adversarial injection prompts and ten matched benign prompts, the deterministic filter alone detected 40.0% of injection attempts (4 of 10; 95% Wilson CI: 16.8% to 68.7%) at a false-positive rate of 20.0% (2 of 10; 95% Wilson CI: 5.7% to 51.0%). Because the filter uses deterministic pattern matching, repeated presentation of an identical prompt returns an identical verdict, so the intervals are computed over the unique prompts rather than over repeated trials, and the small size of the prompt set is why they

How to cite this article:

are wide. These figures describe the deterministic layer in isolation. Because a pre-filter hit blocks a message before the policy evaluator is called, a false positive at this stage is not recoverable downstream, which is a direct usability cost and a further reason why the layer is positioned as a supplement rather than a primary defense. The primary defense against injection is the LLM policy layer behind it, and a full end-to-end evaluation of the two layers combined against a large, off-list adversarial corpus is within the scope of future work. The architectural contribution is the defense-in-depth arrangement, a zero-cost deterministic screen in front of the model, not a claim that the screen alone is sufficient.

Discussion

The results jointly support one claim: an adaptable, multimodal, pre-delivery moderation system can be assembled from interchangeable off-the-shelf models plus an architecture, with no training in the loop. The gatekeeper's structural enforcement closes the timing gap by construction. The policy-content result shows that the core promise of Policy-as-Prompt, adaptation without retraining, produces a real and significant recall gain rather than a theoretical one. The cross-policy result shows the same mechanism carrying across fifteen different policies with no retraining, the property that makes the approach deployable across teams. The sensitivity result shows a second administrator-facing control with a measurable, interpretable, asymmetric effect. The latency result shows that the live system operates within real-time bounds and provides the empirical basis for the plugin architecture's interchangeability claim. Together, these results describe a system whose novelty is the arrangement, not any single model.

Table 7 positions Convo-Ease against representative systems across the dimensions that matter for enterprise deployment. Text-only guardrails are strong on speed and configurability but provide no image or audio coverage and no organizational context [1,2]. Multimodal systems extend coverage but pay for it in latency and stop short of audio support or natural-language policy adaptation [5,6,9]. None combines real-time text, image, and audio coverage with adaptable natural-language policy and pre-delivery enforcement. Commercial cloud moderation services share the same structural limits: predefined, non-extensible rules; text-dominant coverage; latency that is unpredictable under chat loads; and a privacy model that requires sending content to an external server, which is incompatible with several legitimate enterprise data-handling requirements. Convo-Ease's plugin-based, on-premises-capable design is built directly against that set of constraints, and its policy-adaptability advantage over a fixed taxonomy is a measured result rather than an assertion.

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

System	Modalities	Real-time	Policy adaptability	Org. focus	Key innovation	Gap addressed
Llama Guard [1]	Text	Yes	Zero-shot taxonomy	AI safety	Configurable safety taxonomy	No multimodal, no org context
LlamaFirewall [2]	Text	Yes	Modular policy engine	Agent security	Layered defense	Limited multimodal, no live chat
Policy-as-Prompt [3]	Text	Yes	Natural-language prompts	Community mod.	Rules as model instructions	No multimodal, no live-chat eval
MM-SOC [5]	Text + Image	No	Benchmark only	Social media	Multimodal benchmark suite	No real-time, no custom policy
HateMM [6]	Text+Image+Audio	Limited	Fixed-task	Hate detection	Transcript-based video fusion	No org context, compute-heavy
Llama Guard 3 Vision [9]	Text + Image	Yes	Zero-shot taxonomy	AI safety	Multimodal Llama Guard	No audio, English-only, 1 image
Convo-Ease (this work)	Text+Image+Audio	Yes	NL rules + intensity, no training	Enterprise chat	Gatekeeper + Policy-as-Prompt + plugin backends	Target niche

TABLE 7: Comparative positioning of Convo-Ease against representative moderation systems across enterprise-deployment dimensions

Table 7 compares systems on the dimensions that decide enterprise fitness, and those dimensions are categorical rather than numerical. A quantitative row-by-row comparison is not available because each system in the table is evaluated on a different corpus under a different task definition, and no two share a benchmark with the present work. The figures that follow are values reported by the cited studies, not measurements made here, and they are not directly comparable to each other or to ours. A leading static-taxonomy guardrail falls to 13.1% F1 when required to enforce free-form user-defined policies [4]. The closest conceptual neighbor, encoding community rules as natural-language instructions, reports median accuracy near 64% across Reddit communities [3]. A multimodal hate-detection framework achieved a macro-F1 score of 0.848 on video against 0.628 area under the receiver operating characteristic curve on image-text memes under one design [6], and a benchmark across social-media tasks finds zero-shot multimodal performance

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. Cureus J Comput Sci 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

consistently weak with a task-dependent modality gap [5]. Read together, these establish the pattern the present architecture is built against: fixed taxonomies degrade sharply when the policy is not the one they were built for, and natural-language policy recovers part of that ground.

The comparison this study can make rigorously is internal, and it is the one the architecture turns on. Every external figure above varies the model, the corpus, and the policy together, so none isolates the contribution of policy content. Holding the model, the items, and the parameters fixed and varying only the policy text recovers 30.7 percentage points of recall and 8.3 points of accuracy (McNemar's exact $p = 0.004$), and that advantage holds in fourteen of the fifteen rule sets rather than resting on a few. Direct benchmarking of an external fixed-taxonomy guardrail on this corpus was considered and not performed because it would require re-expressing fifteen organizational policies as taxonomy categories, which is precisely the transformation the architecture exists to avoid; a comparison run that way would measure the quality of that translation rather than either system. Evaluating an external guardrail under its own taxonomy on this corpus, alongside a policy-specific annotation of the same messages, is the sound form of that comparison and is listed in future work.

The modularity supports genuinely different domains through policy edits alone. In education, the system can enforce academic-integrity policies and constrain assignment sharing around deadlines. In corporate settings, it can enforce information-classification policies, professional-communication norms, and jurisdiction-specific compliance. In healthcare, it can govern patient-related discussions and enforce the boundary between non-clinical channels and supervised clinical advice. In each case, the change is to the natural-language policy, not to the model.

Limitations and future work

The limitations are stated plainly. The system inherits the failure modes of its underlying models: occasional hallucination, sensitivity to prompt phrasing, a finite context window, and the inference cost of a large model. The evaluation isolates the policy-content variable well, but is run on a proxy corpus built from public labeled data; real organizational chat has a distribution that the system will meet only in deployment. The proxy construction has a specific consequence that bounds what the benchmark can show. Rule sets were paired with messages at random, and ground-truth labels were carried over from the source corpora's toxicity and hate-speech annotations rather than re-annotated against the assigned organizational rule. The per-rule cells contain between 15 and 32 messages, which gives the per-rule comparisons in Table 3 limited power individually, so the cross-policy findings are stated at the level of the set of fifteen rather than for any single rule set, and a larger per-rule sample would be required to characterize any individual policy. A source label is a reliable indicator of whether a message is harmful in the general sense the original annotators applied, but it is not a policy-specific judgment about the particular rule the message was evaluated under. Absolute accuracy figures on this benchmark should be read with that in mind. The paired comparisons are less affected because the same pairing and the same labels are used in every condition, so label mismatch is constant across conditions and cannot generate a difference between them. Policy-specific human annotation against each organizational rule set and replication on real organizational chat are the measurements that would establish absolute performance, and both remain future work. The multimodal claim is architectural: this study measures the text policy-evaluation core directly but does not report end-to-end accuracy on real image and audio media, and per-modality latency for the captioning and transcription stages is likewise not yet profiled on real files. The deterministic pre-filter's standalone injection coverage is limited by design, and a combined two-layer robustness evaluation against a large off-list adversarial corpus remains to be run. All reported results come from a single hosted model in a single benchmark run, so they characterize this architecture with this evaluator rather than establishing model-independent generalization. The plugin architecture makes substituting the evaluator a configuration change, but replication across additional models has not been performed. Finally, evaluating against a hosted third-party model exposes a benchmark to that provider's variability, which the system's fail-closed design protects a deployment against but which complicates a perfectly controlled measurement.

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

Future work follows three horizons. In the near term, the immediate open items are end-to-end multimodal measurement, real images through the captioner, and real audio through transcription scored against labels; per-modality latency profiling of those stages; and a combined-layer adversarial robustness study. In the medium term, prompt decomposition for compositionally complex policies is the natural accuracy improvement consistent with the no-training design, and video moderation can be added by sampling keyframes through the existing image path with no change to the rest of the pipeline; the plugin architecture also admits optional parameter-efficient fine-tuning [21,22] for organizations that want domain specialization beyond prompt-level policy, which we leave to future work as an extension rather than a requirement. In the long term, deeper explainability with image-region attribution grounded in attribution methods [23,24], user-configurable thresholds alongside organizational policy, longitudinal policy-trend analysis [25], and cross-channel consistency across email, chat, and voice would extend the framework into a uniform organizational moderation layer.

Conclusions

Convo-Ease is an architecture for real-time multimodal moderation in enterprise communication, and a working system that realizes it. Its contribution is the arrangement of four design decisions: a pre-delivery gatekeeper that enforces validation structurally in the send path, a Policy-as-Prompt mechanism that adapts to organizational rules with no training, a plugin-based backend that makes any per-modality model interchangeable through configuration, and a unified text abstraction that lets one policy layer govern text, image, and audio. The evaluation reported here measures the text policy-evaluation core; the image and audio paths are established architecturally but are not validated end-to-end on real media. The central claim is validated under controlled, paired conditions: on the same model and the same 300 messages, reading organizational rules instead of a generic safety taxonomy raises recall by more than thirty percentage points, a statistically significant difference, and the same mechanism carries fifteen distinct policies with no retraining. A second control over enforcement intensity has a measurable, asymmetric, and therefore actionable effect, and the live system evaluates policy within real-time bounds. Convo-Ease shows that the enterprise moderation problem is solvable as a systems problem by arranging interchangeable off-the-shelf models behind an adaptable policy layer and a structural gate, rather than by training a new model, and that the resulting design occupies a position that no existing system fills.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Soham S. Jadhav, Omkar N. Gadakh, Nisha D. Patil

Acquisition, analysis, or interpretation of data: Soham S. Jadhav, Atharv Gaikwad

Drafting of the manuscript: Soham S. Jadhav, Omkar N. Gadakh, Atharv Gaikwad

Critical review of the manuscript for important intellectual content: Soham S. Jadhav, Omkar N. Gadakh, Nisha D. Patil

Supervision: Nisha D. Patil

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue.

Animal subjects: All authors have confirmed that this study did not involve animal subjects or tissue.

Conflicts of interest: In compliance with [the ICMJE uniform disclosure form](#), all authors declare the following:

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. *Cureus J Comput Sci* 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

- **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work.
- **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work.
- **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

Data Availability Statements

The data (and/or code) supporting this study are openly available in Zenodo (Convo-Ease v1.0.0) at 10.5281/zenodo.21128761, reference 21128761.

References

1. Inan H, Upasani K, Chi J, et al.: [Llama Guard: LLM-based input-output safeguard for human-AI conversations \[PREPRINT\]](#). arXiv. 2023, [10.48550/arXiv.2312.06674](#)
2. Chennabasappa S, Nikolaidis C, Song D, et al.: [LlamaFirewall: an open source guardrail system for building secure AI agents \[PREPRINT\]](#). arXiv. 2025, [10.48550/arXiv.2505.03574](#)
3. Palla K, Redondo García JL, Hauff C, et al.: [Policy-as-Prompt: Rethinking content moderation in the age of large language model](#). FAccT '25: Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency. 2025, 840-854. [10.1145/3715275.3732054](#)
4. Hoover M, Baherwani V, Jain N, et al.: [DynaGuard: a dynamic guardian model with user-defined policies \[PREPRINT\]](#). arXiv. 2025, [10.48550/arXiv.2509.02563](#)
5. Jin Y, Choi M, Verma G, Wang J, Kumar S: [MM-Soc: benchmarking multimodal large language models in social media platforms \[PREPRINT\]](#). arXiv. 2024, [10.48550/arXiv.2402.14154](#)
6. Koushik GA, Kanojia D, Treharne H: [Towards a robust framework for multimodal hate detection: a study on video vs. image-based content](#). WWW '25: Companion Proceedings of the ACM on Web Conference 2025. 2025, 2014-2023. [10.1145/3701716.3718382](#)
7. Kearns L: [Content moderation assistance through image caption generation](#). Intelligent Systems with Applications. 2025, 25:200489. [10.1016/j.iswa.2025.200489](#)
8. Gou Y, Chen K, Liu Z, et al.: [Eyes closed, safety on: protecting multimodal LLMs via image-to-text transformation](#). Computer Vision - ECCV 2024. Leonardis A, Ricci E, Roth S, Russakovsky O, Sattler T, Varol G (ed): Springer, Cham; 2025. 15075:388-404. [10.1007/978-3-031-72643-9_23](#)
9. Chi J, Karn U, Zhan H, et al.: [Llama Guard 3 Vision: safeguarding human-AI image understanding conversations \[PREPRINT\]](#). arXiv. 2024, [10.48550/arXiv.2411.10414](#)
10. Luo W, Ma S, Liu X, Guo X, Xiao C: [JailBreakV: a benchmark for assessing the robustness of multimodal large language models against jailbreak attacks \[PREPRINT\]](#). arXiv. 2024, [10.48550/arXiv.2404.03027](#)
11. Kang M, Chen Z, Xu C, et al.: [Poly-Guard: massive multi-domain safety policy-grounded guardrail dataset \[PREPRINT\]](#). arXiv. 2026, [10.48550/arXiv.2506.19054](#)
12. Radford A, Kim JW, Xu T, Brockman G, McLeavey C, Sutskever I: [Robust speech recognition via large-scale weak supervision](#). Proceedings of the 40th International Conference on Machine Learning. 2023, 28492-28518.
13. Graham C, Roll N: [Evaluating OpenAI's Whisper ASR: Performance analysis across diverse accents and speaker traits](#). JASA Express Letters. 2024, 4:025204. [10.1121/10.0024876](#)
14. Li J, Li D, Savarese S, Hoi S: [BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models](#). Proceedings of the 40th International Conference on Machine Learning. 2023, 19730-19742.
15. Guo W, Wang J, Wang S: [Deep multimodal representation learning: a survey](#). IEEE Access. 2019, 7:63373-63394. [10.1109/access.2019.2916887](#)

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. Cureus J Comput Sci 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>

16. Brown TB, Mann B, Ryder N, et al.: [Language models are few-shot learners](#). Proceedings of the 34th International Conference on Neural Information Processing Systems. 2020, 1877-1901.
17. [Toxic Comment Classification Challenge](#). (2018). Accessed: November 10, 2025: <https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge>.
18. de Gibert O, Perez N, García-Pablos A, Cuadros M: [Hate speech dataset from a white supremacy forum](#). Proceedings of the 2nd Workshop on Abusive Language Online (ALW2). 2018, 11-20. [10.18653/v1/W18-5102](#)
19. [Nvidia NIM: gpt-oss-120b](#). (2026). Accessed: February 18, 2026: <https://build.nvidia.com/openai/gpt-oss-120b>.
20. [Jadhav S. sohamjadhav95/Convo-Ease: Convo-Ease v1.0.0 \(paper release\)](#). Zenodo. (2026). Accessed: February 18, 2026: <https://doi.org/10.5281/zenodo.21128761>.
21. Hu EJ, Shen Y, Wallis P, et al.: [LoRA: low-rank adaptation of large language models \[PREPRINT\]](#). arXiv. 2021, [10.48550/arXiv.2106.09685](#)
22. Dettmers T, Pagnoni A, Holtzman A, Zettlemoyer L: [QLoRA: efficient finetuning of quantized LLMs](#). 37th Conference on Neural Information Processing Systems (NeurIPS 2023). 2023, 1-28.
23. Lundberg SM, Lee S-I: [A unified approach to interpreting model predictions](#). NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017, 4768-4777.
24. Rodis N, Sardinios C, Radoglou-Grammatikis P, Sarigiannidis P, Varlamis I, Papadopoulos GT: [Multimodal explainable artificial intelligence: a comprehensive review of methodological advances and future research directions \[PREPRINT\]](#). arXiv. 2024, [10.48550/arXiv.2306.05731](#)
25. Nedungadi P, Veena G, Tang K-Y, Menon RRR, Raman R: [AI techniques and applications for online social networks and media: insights from BERTopic modeling](#). IEEE Access. 2025, 23:37389-37407. [10.1109/access.2025.3543795](#)

How to cite this article:

Jadhav S S, Gadakh O N, Gaikwad A, et al. (September 09, 2026) Convo-Ease: A Policy-as-Prompt Gatekeeper for Real-Time Multimodal Moderation in Enterprise Communication. Cureus J Comput Sci 3 : es44389-026-00270-0. DOI <https://doi.org/10.7759/s44389-026-00270-0>