

# Context-Aware Evidence Minimization for Privacy-Conscious Harmful-Speech Dataset Release

Batın Örene

Department of Computer Engineering  
Bahçeşehir University, Istanbul, Türkiye  
batin.orene@bahcesehir.edu.tr

## Abstract

Harmful-speech datasets drive moderation research but are risky to release as raw text. We ask how much source-text exposure curators can remove while preserving evidence for model comparison and error analysis. Releasing only classifier-important tokens corrupts the data: an independent, lexicon-free LLM judge recovers the targeted group from such excerpts only 36% of the time, and reads 53% of non-hateful excerpts as hateful. We introduce I-CEM, a deterministic evidence-minimization baseline that masks recorded synthetic identifiers, retains tokens whose removal lowers a frozen classifier’s harmful-speech score, and expands retained spans only when rules detect target, harm, negation, quotation, or counter-speech context. We evaluate on HateXplain, HateCheck, and Civil Comments with controlled synthetic PII injection. Our primary result is target recovery: I-CEM raises it from 58% to 67% over a token-matched fixed window ( $p < .001$ ), and the gain replicates against the dataset’s gold target labels (50% to 56%,  $p < .001$ ); I-CEM also more than halves the importance-only misreading (25%). Secondary gains are directional: hate-label fidelity improves modestly (90% vs. 88%,  $p = .04$ ) but does not hold against gold labels, and the stance advantage over a matched window is not significant. On 5,000 HateXplain examples I-CEM retains about half of source tokens and stays within 0.02 positive-class F1 of a fixed importance window, but releases more text than that window when its context rules fire, a trade curators opt into. Masking every recorded injected identifier validates the masking pipeline; it is not evidence of real-world anonymization. I-CEM is a deterministic context-recovery baseline that measurably reduces label corruption from evidence-only release.

## 1 Introduction

Harmful-speech datasets often originate in user-generated posts and comments (Waseem and Hovy,

2016; Davidson et al., 2017; Mathew et al., 2021).<sup>1</sup> A decade of such dataset creation has made re-releasing and re-sharing these corpora a routine part of moderation research, turning disclosure risk from an edge case into a recurring curation decision. Sharing such text is useful for reproducibility, error analysis, and model comparison, but it can also preserve direct identifiers, quasi-identifiers, and more surrounding context than is needed for model comparison or error analysis. Text anonymization work frames this as a disclosure-risk and utility tradeoff (Lison et al., 2021; Pilán et al., 2022). For harmful-speech data, that tradeoff is especially awkward: removing too much context can erase the target, quoted stance, negation, or counter-speech cues that determine whether a text is hateful.

We use *release minimization* to mean replacing a raw example with a shorter masked excerpt before sharing it. The central question is: how much source text can be removed while preserving classifier utility, the target of harm, cues about whether the harm is asserted, quoted, negated, or opposed, and low residual exposure of identifiers? We avoid paraphrase generation and instead release masked excerpts made from original token spans, so each output remains traceable to the source tokens. We do not ask whether this produces anonymized data. Instead, we ask what a curator can release when the goal is to expose less text while preserving enough evidence for model comparison and error analysis.

We introduce I-CEM, an importance-guided context-aware evidence minimization baseline. I-CEM masks recorded synthetic direct/quasi identifier spans and deterministic identifier matches, selects non-identifier evidence tokens by measuring how much each token changes a frozen classifier’s harmful-speech score, and expands around selected tokens only when deterministic context

<sup>1</sup>Content warning: this paper concerns hateful content. All examples are synthetic and use placeholders; no real posts, slurs, or profanity are reproduced.

rules identify nearby target, harm, negation, quotation/reporting, or counterspeech cues. This sits between full-text PII masking, which exposes all non-PII text, and importance-only release, which preserves model signal but destroys review-relevant target and stance context. The contribution is therefore not a new hate-speech classifier or a complete de-identification system; it is a deterministic release baseline and evaluation protocol for measuring identifier residuals, utility, compression, and context tradeoffs separately.

Our experiments show that I-CEM masks all recorded injected identifiers in the main controlled setting and retains about half of source tokens. An independent, lexicon-free LLM judge shows what the naive alternative costs: releasing only classifier-evidence tokens corrupts the label, making a reader misread most counter-speech and quoted content as hateful and lose the target of harm. We frame these results as evidence for a deterministic release protocol, not as a formal privacy guarantee.

We order our findings by how strongly they are grounded. Target recovery is the primary result: I-CEM significantly improves recovery of the targeted group over a token-matched fixed window, and it is the one effect that replicates under both references we have, an independent judge and the dataset’s gold target labels. Next is the corruption finding above: every form of context recovery repairs a large part of it. The advantages over a *wide* matched window on hate-label fidelity and on stance are directional only, and when I-CEM’s rules fire it releases more text than a fixed window would, so the added context is a trade a curator chooses rather than a free improvement.

## 2 Related Work

Early Twitter hate-speech datasets established common harmful-speech evaluation settings (Waseem and Hovy, 2016); Davidson et al. show the difficulty of separating hate speech from offensive but non-hateful content (Davidson et al., 2017). HateXplain adds target communities and human rationales, making it useful for studying whether reduced texts still preserve review-relevant evidence (Mathew et al., 2021). HateCheck complements held-out corpus evaluation with 29 functionality tests designed to expose specific model weaknesses (Röttger et al., 2021). Borkan et al. release roughly two million online comments labeled for toxicity, a subset further annotated by crowd workers for

the identities they reference, supporting threshold-agnostic measurement of unintended bias (Borkan et al., 2019).

I-CEM’s evidence step is an instance of erasure- or occlusion-based attribution, which scores input units by the change in model output when they are removed (Li et al., 2016). Releasing only such evidence connects to extractive rationalization, where a selector keeps a short input subset sufficient for the prediction (Lei et al., 2016; Jain et al., 2020); the ERASER benchmark formalizes evaluating such rationales against human annotations (DeYoung et al., 2020), and we reuse HateXplain’s rationales for the same purpose. Unlike rationalization, our goal is not to explain a prediction but to decide what text a curator can release, so we add deterministic context expansion and measure label corruption rather than faithfulness. A separate line perturbs or paraphrases text under (metric) differential privacy (Feyisetan et al., 2020); we instead release verbatim token spans so each output stays traceable to the source, a property such perturbation does not provide and whose word-level guarantees have known limits (Mattern et al., 2022).

Text anonymization and privacy-preserving NLP have emphasized pairing utility measurement with explicit privacy-risk measures rather than relying only on downstream task accuracy (Coavoux et al., 2018; Sousa and Kern, 2023). TAB explicitly evaluates both privacy protection and utility preservation for text anonymization (Pilán et al., 2022); recent work further surveys the diversity of privacy metrics and cautions that different notions capture different risks (Ren et al., 2025). Pseudonymization studies also show that anonymization choices can alter downstream NLP behavior (Yermilov et al., 2023). Our work applies this privacy-utility viewpoint to harmful-speech dataset release, where context loss can change label interpretation.

## 3 Method

Given an input example  $x$ , I-CEM produces a released text  $r$  through four steps.

**Step 1: Identifier detection.** First, it detects direct identifiers (such as names, handles, and contact strings) and quasi identifiers (such as dates, locations, schools, and workplaces). In these controlled experiments, this stage uses gold spans from the synthetic PII injection plus deterministic regular-expression and phrase rules; a real release would require a stronger PII recognizer.

**Step 2: Classifier scoring.** Second, it scores the full text with a frozen harmful-speech classifier.

**Step 3: Token importance.** Third, it computes token importance. Let  $s(x)$  be the harmful-class score and let  $x_{-i}$  be the text with token  $i$  replaced by [MASKED]. We score eligible tokens by  $\Delta_i = s(x) - s(x_{-i})$ , i.e. leave-one-out erasure attribution (Li et al., 2016). Eligible tokens cannot overlap identifier spans, placeholders, punctuation-only tokens, or common stopwords, except for context words such as negation and reporting cues.

**Step 4: Release rendering.** Fourth, it renders one of several release variants.

The *importance-only* variant keeps only selected evidence tokens. The *importance-window* variant keeps a symmetric token window around each selected evidence token. I-CEM also starts with symmetric windows around the same selected evidence tokens.

I-CEM adds longer retained spans only when deterministic lexical rules identify review-relevant context. When a target cue and a harm cue occur within eight tokens, I-CEM keeps the text between them; when negation, quotation/reporting, or counterspeech cues occur within eight tokens of evidence or harm cues, it also keeps that local relation. Thus the difference from the fixed-window baseline is not the evidence selector, but the extra context retained when target-harm or stance-harm relations are detected. The gain is therefore in *which* tokens are kept: relation-anchored spans rather than symmetric padding, not simply more tokens released, as the token-matched comparison in Table 1 confirms. For instance, given source [GROUP] [SLUR] is wrong; I do not hold that view, importance-only keeps only the two highest- $\Delta_i$  tokens ([GROUP] [SLUR]), stripping the negation; the fixed window adds symmetric padding ([GROUP] [SLUR] is wrong); I-CEM detects the negation cue within the evidence window and expands to [GROUP] [SLUR] is wrong ... do not hold that view. The cue classes are fixed lexicons matched on lowercased tokens. Target cues are identity-group references (and any per-example target labels the dataset provides), harm cues are hostile-action verbs and slurs, and stance cues are negation, quotation/reporting, and counterspeech markers. Appendix B lists the inventories;

---

**Algorithm 1** I-CEM release of one example

---

**Require:** text  $x$ ; frozen classifier  $s$   
**Require:**  $\text{top\_k}$ , radius  $r$ , threshold  $\tau=0.02$ , cap  $c=0.30$ , gap  $g=8$   
**Ensure:** released text  $\hat{x}$

- 1:  $I \leftarrow$  direct/quasi identifier spans in  $x$
- 2:  $\Delta_i \leftarrow s(x) - s(x_{-i})$  for each eligible token  $i \notin I$
- 3:  $A \leftarrow$  top- $k$  tokens with  $\Delta_i \geq \tau$ , capped at  $c|x|$
- 4: **if**  $A = \emptyset$  **then**
- 5:    $A \leftarrow \{\arg \max_i \Delta_i\}$     $\triangleright$  fallback: take strongest token
- 6: **end if**
- 7:  $R \leftarrow \bigcup_{i \in A} \text{WINDOW}(i, r)$     $\triangleright$  evidence windows
- 8: **for** each target cue  $t$ , harm cue  $h$  with  $|t - h| \leq g$  **do**
- 9:    $R \leftarrow R \cup \text{SPAN}(t, h)$     $\triangleright$  target-harm bridge
- 10: **end for**
- 11: **for** each stance cue  $u$  within  $g$  of any anchor or harm cue  $v$  **do**
- 12:    $R \leftarrow R \cup \text{SPAN}(u, v)$     $\triangleright$  stance-harm bridge
- 13: **end for**
- 14: Mask  $I$ ; render  $R$ , joining gaps with omission markers
- 15: **return**  $\hat{x}$

---

following WOA guidance<sup>2</sup> we describe and count the harm-cue slurs rather than reprint them as a dictionary, and release the verbatim list only in the released code. We select up to  $\text{top\_k}$  eligible tokens with  $\Delta_i \geq .02$ , capped at 30% of source tokens. If no token passes the threshold, the selector falls back to the strongest eligible non-identifier token. This fallback avoids empty releases for examples with weak but nonzero model evidence. All reduced variants mask direct and quasi identifiers before rendering; discontinuous retained spans are joined with omission markers rather than paraphrased. Appendix A gives synthetic short, medium, and long examples of the resulting releases; no real posts, slurs, or profanity are reproduced. Algorithm 1 summarizes the full procedure.

We use  $\text{top\_k}=5$  and  $\text{window\_radius}=2$  as the main setting. A grid sensitivity sweep shows this is a balanced point rather than the most favorable setting for any single metric. The remaining hyperparameters (the importance threshold  $\Delta \geq .02$ , the eight-token target-harm and stance-harm gaps, and the 30% anchor cap) were fixed a priori as conservative defaults and not swept; only  $\text{top\_k}$  and  $\text{window\_radius}$  are tuned, which keeps I-CEM a deterministic baseline rather than a heavily optimized system.

---

<sup>2</sup><https://www.workshopononlineabuse.com/policies.html>

## 4 Experimental Setup

We evaluate on HateXplain (Mathew et al., 2021) as the main harmful-speech benchmark, HateCheck (Röttger et al., 2021) as a functional context stress test, and the `google/civil_comments` release of the Jigsaw Unintended Bias/Civil Comments toxicity data (Borkan et al., 2019; Google Research, 2024) as an out-of-domain sensitivity check. To validate our context proxies against human judgments on a second, independent dataset, we also use SemEval-2021 Toxic Spans Detection (Pavlopoulos et al., 2021; Toxic Spans Detection (SemEval-2021 Task 5), 2021), whose human-annotated toxic character spans give a gold rationale signal with a different annotation scheme from HateXplain. Because these datasets provide task labels, and in some cases context annotations, rather than gold PII spans, we add deterministic synthetic PII and record gold direct and quasi identifier spans for residual-span measurement. For each row, a seed plus row index selects one of six clearly fake insertion templates: a handle prefix, person-school prefix, email suffix, date-person-location prefix, family-person-phone prefix, or workplace-event suffix. The injector records exact character offsets, labels, and placeholder replacements for every inserted span; these recorded spans are the gold direct/quasi identifiers used by the residual metric. For example, if  $x$  denotes the original row text, one injected row may be `On 2026-04-17, [PERSON] in [LOCATION] posted: x`, with `DATE`, `PERSON`, and `LOCATION` recorded as gold spans. This creates a controlled comparison of identifier residuals across release variants; it does not model all real-world disclosure risk.

We compare six variants: raw text, direct-PII masking, direct+quasi masking, importance-only evidence, fixed importance windows, and I-CEM. We report positive harmful-class F1 on released text (F1+), prediction flip rate relative to raw text (Flip), retained source-token percentage (Tok), and residual rate for recorded injected identifier spans (PII). We binarize the classifier’s harmful score at a fixed 0.5 decision threshold for both raw and released text. F1+ is the positive-class F1 of the released-text decision against the dataset’s binary label; Flip is the fraction of examples whose binarized decision changes between raw and released text. Tok is the percentage of source surface tokens that fall inside a retained span, where tokens come

from a deterministic regular-expression tokenizer over words, numbers, URLs, handles, and placeholders, not the classifier’s subword units. The classifier always scores the rendered release string, including masks and omission markers, exactly as a downstream user would receive it. Target is the fraction of examples with an automatic source target cue whose release retains at least one target cue. Stance is the fraction of examples with automatic source negation, quotation/reporting, or counter-speech cues whose release retains both a stance cue and a harm cue. These context metrics are lexical proxies, not human judgments. The Target and Stance proxies use the same fixed cue lexicons that I-CEM’s context rules use to decide expansion (Appendix B); they therefore measure whether the method’s own context relations survive in the release, not whether a human reviewer would judge the text sufficient. We read high Target and Stance for I-CEM as by-construction context recovery, and validate it externally: gold rationales annotate the target span, and an independent lexicon-free LLM judge (Appendix E.1) grounds the stance result, which the cue lexicons cannot annotate. To validate the target proxy against a human signal, we additionally report Rationale overlap: the fraction of HateXplain’s crowd-sourced rationale tokens (the tokens annotators marked as responsible for the label) that the release retains. PII residuals are also checked with deterministic residual detectors. In the main run, the same DeHateBERT model selects evidence tokens and scores released text (Aluru et al., 2021; Hate-speech-CNERG, 2020). To reduce the concern that selection and evaluation use the same model, we also rescore the saved HateXplain releases with a Detoxify toxicity model (Hanu and Unitary team, 2020).

All Table 1 values come from fixed saved runs, each with its config and manifest in the public release, which contains aggregate metrics, configs, manifests, and code, but not raw dataset rows or transformed real-text row dumps. Code, run configs, run manifests, and aggregate results are publicly available at <https://github.com/batinium/icem-woah2026> and archived at <https://doi.org/10.5281/zenodo.21974841>. All four datasets are English. Both classifiers are frozen base-size transformers used for inference only, with no training or fine-tuning: DeHateBERT (BERT-base scale, ~110M parameters) and Detoxify unbiased-toxic-roberta

(RoBERTa-base scale,  $\sim 125\text{M}$  parameters). All runs are inference on a single consumer Apple-silicon GPU (PyTorch MPS backend); no model parameters are updated. Library versions are pinned in the public release.

## 5 Results

Table 1 shows the main tradeoff. The importance-only variant releases very little text, but it loses most target and stance context. A fixed importance window ( $r=2$ ) recovers some context and has the highest F1, but still preserves only 24.6% of stance-harm relations while releasing fewer tokens than I-CEM (39.8% vs. 49.7%). Because that baseline operates at a smaller token budget, a context comparison against it would conflate *how much* text is released with *which* text. We therefore also report a token-matched fixed window ( $r=3$ ) that releases 48.8% of source tokens, nearly matching I-CEM’s 49.7% (the radius was chosen post hoc to match I-CEM’s token budget, not tuned for context). I-CEM’s advantage is not that it releases more tokens, but that its relation-anchored spans capture target and stance context that symmetric padding misses at the same exposure. Even at this matched budget, the fixed window preserves far fewer of the method’s context relations than I-CEM: target preservation rises only to 77.2% (vs. 95.3%) and stance to 33.8% (vs. 61.9%), with a higher flip rate (15.0% vs. 13.4%). These are cue-preservation proxies scored with I-CEM’s own lexicons, and the independent judge later shrinks the matched-budget stance gap to a non-significant difference, so the proxy multiples in this table should be read as an upper bound on the effect rather than its size. We caution that positive-class F1 is not monotone in reduction: importance-only (.704) and the  $r=2$  window (.716) exceed even raw text (.684), because discarding context raises positive-class recall. Macro-F1, which is not inflated this way, declines monotonically from raw (.645) to I-CEM (.636), the  $r=2$  window (.633), and importance-only (.604). We therefore read Flip, not F1+, as the primary utility signal, and treat the small F1+ gaps as a secondary check rather than evidence that reduction preserves utility for free. Compared with direct+quasi masking, I-CEM releases roughly half as many source tokens while preserving most target cues and masking all recorded injected direct/quasi identifier spans. The PII column is zero by construction: in these controlled runs the masker re-

| Variant                                   | F1+  | Flip | Tok   | PII | Target | Stance |
|-------------------------------------------|------|------|-------|-----|--------|--------|
| <i>Main DeHateBERT selector/evaluator</i> |      |      |       |     |        |        |
| PII+quasi mask                            | .692 | 8.5  | 100.0 | 0.0 | 100.0  | 77.6   |
| Importance only                           | .704 | 26.7 | 11.7  | 0.0 | 54.5   | 4.5    |
| Importance window ( $r=2$ )               | .716 | 18.0 | 39.8  | 0.0 | 72.2   | 24.6   |
| Importance window ( $r=3$ )               | .703 | 15.0 | 48.8  | 0.0 | 77.2   | 33.8   |
| I-CEM                                     | .697 | 13.4 | 49.7  | 0.0 | 95.3   | 61.9   |
| <i>Independent Detoxify evaluator</i>     |      |      |       |     |        |        |
| Importance only                           | .674 | 21.3 | 11.7  | –   | –      | –      |
| Importance window ( $r=2$ )               | .693 | 14.5 | 39.8  | –   | –      | –      |
| I-CEM                                     | .694 | 12.9 | 49.7  | –   | –      | –      |

Table 1: HateXplain, 5,000 examples. Flip: prediction flip rate vs. raw text. Tok: retained source-token %. PII: injected-identifier residual % (zero by construction; the masker receives recorded gold spans, confirming pipeline correctness, not anonymization; – = independent evaluator reuses same releases, no separate PII metric). Target/Stance: cue-preservation proxies (Stance requires a stance cue *and* a harm cue); identical across blocks since the independent evaluator changes only F1+ and Flip.  $r=3$  window is a token-matched baseline for I-CEM, isolating *which* tokens are kept from *how many*. Detoxify block omits PII+quasi mask (100% tokens, replicates raw-text utility).

ceives the same recorded gold spans the injector planted, so a 0.0% residual confirms that the masking pipeline executes, not that an imperfect detector would catch real-world identifiers. We treat this column as a plumbing check and discuss the consequence in the Limitations.

Under the independent evaluator, I-CEM matches the fixed window on F1 while preserving higher target and stance context rates (Table 1). This suggests that the pattern is not merely an artifact of evaluating DeHateBERT-selected spans with DeHateBERT.

HateCheck supports the context interpretation. Denominators differ because a functionality group counts toward a metric only when its cases contain that cue: all 29 define flips and quotation, but only 26 reference a target and 22 a stance-harm relation. Across the groups where each metric is defined, I-CEM improves on the  $r=2$  fixed window in target preservation for 23/26 functionality groups, stance-harm preservation for 16/22, negation preservation for 22/29 and quotation preservation for 29/29, and lowers raw-decision flips in 17/29 groups (against importance-only the corresponding counts are 26/26, 22/22, 27/29, 29/29, and 26/29). This comes with higher retained-token exposure and is therefore best interpreted as a controlled context-recovery tradeoff. Unlike Table 1, this HateCheck comparison is not token-budget-matched, so part of each group-level gain reflects

| Setting                                                 | Variant      | F1+  | Flip | Tok  | Target | Stance |
|---------------------------------------------------------|--------------|------|------|------|--------|--------|
| <i>Detoxify (unbiased-toxic-roberta) selector (014)</i> |              |      |      |      |        |        |
|                                                         | Window (r=2) | .723 | 16.6 | 41.4 | 80.6   | 28.4   |
|                                                         | I-CEM        | .716 | 13.5 | 50.9 | 97.0   | 63.9   |
| <i>No synthetic PII injection (015)</i>                 |              |      |      |      |        |        |
|                                                         | Window (r=2) | .726 | 12.2 | 49.7 | 71.9   | 37.4   |
|                                                         | I-CEM        | .730 | 10.1 | 57.5 | 93.8   | 81.4   |

Table 2: Robustness checks. The context-recovery pattern from Table 1 persists when a different classifier selects evidence tokens (014) and when no synthetic identifiers are injected (015). Columns match Table 1; the independent-evaluator check (Detoxify) is reported in Table 1.

I-CEM’s higher token exposure rather than better span selection; we therefore read these counts as directional support, not a budget-controlled result.

Two further runs test whether the pattern survives changes to the pipeline (Table 2). Replacing the DeHateBERT token selector with the same Detoxify `unbiased-toxic-roberta` model used as our independent evaluator (run 014) keeps the qualitative result: I-CEM lowers the flip rate relative to the fixed window and roughly doubles stance preservation, with target preservation near ceiling. Repeating the main HateXplain run without synthetic PII injection (run 015), which tests whether the effect depends on planted identifiers, shows the same direction. In both cases I-CEM again retains more text than the fixed window, so these are robustness checks on the context–exposure tradeoff rather than free improvements. The independent Detoxify evaluator (Table 1) is the third such check.

Because our target and stance metrics are lexical proxies, we ground them in an external human signal: HateXplain’s crowd-sourced rationales, which mark the tokens annotators judged responsible for the label (using no-PII runs, where the rationale mask stays aligned). I-CEM retains far more human-flagged content than importance-only and as much as a matched window, confirming the proxies track genuinely label-relevant content. Over four seeds, I-CEM retains  $75.7 \pm 0.5\%$  of gold rationale tokens versus  $68.8 \pm 0.5\%$  for the fixed window and  $36.9 \pm 0.2\%$  for importance-only; at a matched budget I-CEM and the window are comparable (75.3% vs. 75.1%). Rationales mark the harmful span, not the stance relation, so we ground the stance proxy separately with an independent, lexicon-free LLM judge that labels the author’s stance toward the harmful content on the full text and on each release (Appendix E.1). On the 228

HateXplain rows whose full text reads as a non-assertion stance, importance-only release preserves that stance in only 1.3% of cases, versus 26.3% for the r=2 window, 42.1% for the token-matched r=3 window, and 49.1% for I-CEM. The judge strongly corroborates the central claim: classifier evidence alone collapses stance and context recovery restores it (all  $p < 10^{-4}$  versus importance-only). It does not corroborate the magnitude of I-CEM’s edge over a token-matched window: the lexical proxy’s  $1.8\times$  matched-budget gap shrinks to about  $1.17\times$  and is not statistically significant (49.1% vs. 42.1%, McNemar  $p = 0.052$ ). We therefore report I-CEM’s advantage over a wide fixed window as directional (consistent with the gold-rationale tie) and treat the large gap over importance-only as the externally validated stance result.

The same independent judge lets us measure what context loss does to the *label* a reader would assign, the outcome that matters for dataset validity (Table 3). Given only an excerpt, the judge decides whether it is hateful and which group it targets; we compare each release to the judge’s reading of the full text over the 963 context-sensitive rows (no-PII, seed 17; rows where the source text contains at least one harm cue and at least one negation, quotation, or counterspeech cue in the shared lexicon). Importance-only release is corrupting: on the 200 rows the full text reads as *not* hateful (counter-speech, negation, quotation), the excerpt flips 53.0% to hateful, and it recovers the target in only 35.5% of cases. The clearest and largest effect is on *target recovery*: I-CEM reaches 67.3% against 58.2% for the token-matched fixed window (McNemar  $p < 10^{-4}$ ), a gap that holds in direction under every robustness check we run and significantly under most of them (gold labels below; three of four judges in Appendix E.2). Hate-label fidelity moves in the same direction but by much less (89.7% vs. 87.7%,  $p = 0.04$ ), and the misread rate falls from 53.0% to 24.5% relative to importance-only while the difference against the matched window is not significant ( $p = 0.31$ ); we treat both as secondary. The stance proxy could not establish the matched-window comparison at all: it is non-significant at the single matched point ( $p = 0.052$ ), though I-CEM does sit 10.5 points above the window frontier interpolated at its own token budget on the exposure–stance curve (Appendix E.1), a descriptive frontier comparison rather than a tested effect. The harm story is concrete: a corpus minimized to classifier evidence alone would system-

atically relabel counter-speech and quoted hate as first-person hate and strip the targeted group. Recovering that context is not free: at this operating point I-CEM releases 57.5% of source tokens where importance-only releases 14.3%, so the comparison a curator faces is target recovery against text exposure, and the matched-window row is the honest reference point for what an equal exposure budget already buys.

**Validating against gold labels.** The judge is the reference for these comparisons, so we check it against the datasets’ own labels, which require no new annotation. On the 963 full texts the judge’s hate decision agrees with the gold binary label 80.0% of the time (recall 91.7%), so it is a usable instrument rather than an arbitrary oracle.

The target-recovery ordering does not depend on the judge: scoring each release’s target against the *gold* HateXplain target groups (with a fixed synonym map; Appendix E.2) reproduces it: importance-only 36.7%, token-matched window 50.3%, I-CEM 56.0%; I-CEM still significantly beats the matched window (McNemar  $p=.0003$ ). The hate-fidelity edge does *not* survive this stricter test (I-CEM 78.1% vs. 76.6%,  $p=.11$ ), consistent with its borderline and judge-dependent significance against the judge reference. We therefore read target recovery as the confirmed, multiply-grounded result and the hate-fidelity edge over a wide matched window as directional. We do not anchor the counter-speech misread metric to gold, because HateXplain labels do not encode stance (a post quoting a slur to condemn it is typically labeled hateful), so the judge’s stance-aware full-text read, not the gold label, is the correct reference there. Across the many paired tests we report we treat as confirmatory only effects that hold under both the judge and the gold reference (or across judges); single  $p$ -values near .05 we label directional.

The same lexical-proxy pattern appears on a second dataset with a different annotation scheme, SemEval-2021 Toxic Spans (CC0): I-CEM preserves 64.2% of stance-harm relations versus 22.5%/30.1% for the  $r=2/r=3$  fixed windows and 1.6% for importance-only, while matching the windows on gold-span overlap (Appendix D). The proxy effect is therefore not HateXplain-specific; Toxic Spans was not independently judged, so the matched-budget caveat above applies: the robust, transferable part is the large gap over importance-

| Variant                   | Tok  | HateFid     | CS→Hate     | TgtRec      |
|---------------------------|------|-------------|-------------|-------------|
| Importance only           | 14.3 | 76.9        | 53.0        | 35.5        |
| Window ( $r=2$ )          | 49.7 | 84.9        | 34.5        | 52.1        |
| Window ( $r=3$ , matched) | 59.7 | 87.7        | 28.0        | 58.2        |
| I-CEM                     | 57.5 | <b>89.7</b> | <b>24.5</b> | <b>67.3</b> |

Table 3: Independent LLM judge on 963 context-sensitive no-PII HateXplain rows (seed 17; Tok values differ from Table 1 because no PII masking is applied). HateFid is agreement of the excerpt’s hate decision with the full-text decision; CS→Hate is the percentage of full-text-non-hateful (counter-speech/quoted) rows the excerpt flips to hateful (lower is better; difference vs. matched window not significant, McNemar  $p = 0.31$ ); TgtRec is target recovery vs. the full text. I-CEM beats the token-matched  $r=3$  window on HateFid ( $p = 0.04$ ) and TgtRec ( $p < 10^{-4}$ ). Details in Appendix E.2.

only, not the precise multiple over a matched window.

## 6 Sensitivity

We treat `top_k` and `window_radius` as positions on a compression-context curve and pick a balanced operating point rather than a metric-maximizing one. A 5,000-example grid sweep over these two parameters (runs 008–012) traces the curve: more compressed settings (`top_k=3` or `radius=1`) release less text but increase raw-decision flips or lose context, while larger settings (`top_k=7`, `top_k=10`, or `radius=3`) release more text for limited additional context. We use `top_k=5`, `radius=2`, where context preservation has largely saturated but token exposure is still moderate. Concretely, across the grid target preservation stays in 93.9–95.6% and stance in 59.9–63.6% while retained tokens move from 43.2% ( $k=5$ ,  $r=1$ ) to 55.8% ( $k=5$ ,  $r=3$ ); the context proxies are nearly flat because they are set by the deterministic context rules rather than the evidence budget. This flatness is consistent with the proxies measuring whether the method’s rules fire (Section 4), and means the operating point mainly trades token exposure and flip rate, not context. A curator could read a different operating point off the same sweep; we do not claim it uniquely optimizes any fixed criterion. Together with the independent evaluator, the selector run with a different classifier, and the no-PII robustness check, the sweep reduces but does not eliminate the risk that the reported setting is tuned to a single favorable measurement.

Rerunning over four seeds (17, 7, 23, 41) confirms the gaps are not seed artifacts: I-CEM target  $95.1 \pm 0.4\%$  and stance  $61.7 \pm 1.0\%$  versus

$72.8 \pm 0.8\%$  and  $24.4 \pm 0.7\%$  for the window, far exceeding the seed noise band ( $\text{std} < 1\text{pt}$ ). F1 differences stay within  $\sim 0.015$  (Appendix C).

## 7 Conclusion

I-CEM is a deterministic release-minimization baseline for privacy-conscious harmful-speech dataset sharing. Importance-only reduction preserves classifier signal while stripping review-critical context; adding targeted context recovers much of that signal while still reducing text exposure. The durable result is target recovery, which improves significantly over an equal-exposure fixed window under both an independent judge and the dataset’s gold target labels; the hate-fidelity and stance advantages over that wide window are weaker and do not survive every check. That context is bought with additional released text whenever the rules fire; I-CEM makes the trade auditable rather than avoiding it. Masking all recorded injected identifiers shows the pipeline works end to end and says nothing about detecting real-world identifiers, which would require a production recognizer and human review.

## Limitations

This work is not a formal privacy guarantee. Synthetic PII enables controlled measurement of residual identifiers, but real-world personal information is broader, contextual, and harder to detect. In particular, the zero PII residual in our tables follows from masking the recorded gold spans that the injector planted; it measures pipeline correctness, not detector skill on unseen identifiers, and a curator must not read a zero-residual score as de-identification. The identifier detector is deterministic and small by design; it is not a production PII recognizer. Token importance is model evidence rather than a human rationale. Target and stance preservation are lexical proxies over fixed cue inventories, and they are scored with the same lexicons that drive I-CEM’s context expansion: they detect whether the method’s own surface cues survive, not whether a human reviewer would judge the released text context-sufficient, and they can both over- and under-count true context. We ground the target proxy in overlap against gold human rationales on two datasets, but those rationales mark label-relevant tokens rather than reviewer-judged context sufficiency, and they do not annotate the stance relation. We instead ground these

proxies in an independent, lexicon-free LLM judge (Appendices E.1, E.2): it confirms that importance-only release collapses stance and corrupts the label, and that I-CEM significantly improves target recovery over a token-matched fixed window. The comparisons the judge does *not* settle are the secondary ones: the single matched-point stance difference is not significant ( $p = 0.052$ ), and the hate-fidelity margin is borderline. I-CEM does sit above the interpolated window frontier at its own exposure, but that is a descriptive comparison across operating points, not a hypothesis test, and the lexical proxy’s matched-budget stance multiple clearly overstates the effect. Because the datasets carry gold task and target labels, we validate the judge against them directly (80% hate agreement with gold) and confirm the target-recovery result holds against gold ( $p=.0003$ ), so the central findings do not rest on the judge alone; the hate-fidelity edge over a wide matched window does not reach significance against gold and is reported as directional. Still, an LLM judge is not a human reviewer (Bavaresco et al., 2025), the gold labels do not annotate stance, and a dedicated multi-rater human study on the release-context question is left to future work. We report descriptive comparisons over fixed saved runs rather than statistical replicability analysis across multiple independently sampled datasets (Dror et al., 2017). Civil Comments is included only as a sensitivity check because the main hate-speech classifier is weak against out-of-domain toxicity labels (raw F1 .125 on 1,000 rows); even there, the context pattern holds, with I-CEM preserving 86.1%/57.7% target/stance context versus 43.9%/17.9% for the fixed window, so we report it as directional support rather than a utility result.

**Context is bought with exposure.** I-CEM intentionally exposes more text than a fixed window whenever its context rules fire: at the main operating point it releases 49.7% of source tokens against 39.8% for the  $r=2$  window, and where we control for this with a token-matched  $r=3$  window the remaining advantage narrows to target recovery. It is therefore not a strictly better point on the compression curve. Curators who cannot afford the extra exposure should read the matched-window rows, not the I-CEM rows, as their baseline.

**Adapting beyond English.** All four datasets and both cue inventories are English, and the context rules fire on surface tokens within a fixed window,

so porting I-CEM to another language is not simply translating the lists. Target and harm cues are the most portable part, since identity terms and slurs are largely lexical, although morphologically rich languages would need lemmatization or subword matching rather than exact lowercased forms. Stance is the hard part: negation, reported speech, and counter-speech are frequently marked by morphology or at the discourse level rather than by the tokens an English list matches. A faithful port would replace the stance lexicon with morphological or dependency-based detection, keeping the deterministic and auditable structure of the rules while changing what they match, and would need a language-appropriate frozen classifier and a judge validated in that language. We did not attempt it here.

**Toward a production identifier recognizer.** The identifier stage in these experiments is deliberately trivial: it receives the recorded gold spans planted by the injector plus a small set of regular-expression and phrase rules, so the zero residual measures plumbing, not recognition. A real deployment would substitute a trained PII recognizer, for example a sequence labeller in the TAB line of work (Pilán et al., 2022), and would then have to report what that recognizer misses rather than a residual of zero. The stages are independent, so recognizer quality and context quality can be measured separately, but they interact in the unsafe direction: context expansion re-releases surrounding text and can surface a quasi-identifier the recognizer failed to flag. The context rules would therefore need to run over recognizer output with a margin, and any serious release would still require human review of the expanded spans. We did not run this experiment.

## Ethics and Data Use

We use existing public research artifacts and cite their creators. Checked source metadata lists HateCheck as CC-BY-4.0, the Hugging Face `google/civil_comments` release as CC0-1.0, and the `heegyu/toxic-spans` mirror of SemEval-2021 Toxic Spans Detection as CC0-1.0. HateXplain metadata is inconsistent across checked sources: current Hugging Face metadata lists CC-BY-4.0, while the dataset-card licensing section and GitHub repository list MIT. Our use is limited to research evaluation, preprocessing, aggregate reporting, and code-level reproducibility. We do not

publish raw posts, real transformed release rows, or new harmful-content examples in the paper or the public release. Following WOA’s guidance on abusive-content examples, the appendix uses synthetic schematic examples with placeholders rather than verbatim social-media content, unobfuscated slurs, or profanity. Synthetic PII is inserted only to create measurable gold spans for a controlled residual metric, and zero residuals in that metric should not be read as real-world de-identification. The main risk we foresee is misuse of that metric: a curator could over-trust a zero-residual score and release under-protected text. We therefore frame I-CEM as a research baseline that reduces exposure, not a de-identification tool, and recommend a production PII recognizer and human review before any real release. The method is privacy-protective by design, so we see limited dual-use potential.

## AI Assistance Disclosure

Claude (Anthropic) assisted with code implementation and debugging, language polishing, and bibliography checks. The author made all research decisions, designed and interpreted the experiments, reviewed all AI-assisted material, and takes full responsibility for the final paper. No AI system is listed as an author.

## Acknowledgements

I thank the anonymous reviewers and the meta-reviewer of WOA 2026, whose comments substantially reshaped how the results in this paper are ordered and qualified.

## References

- Sai Saketh Aluru, Binny Mathew, Punyajoy Saha, and Animesh Mukherjee. 2021. [A deep dive into multi-lingual hate speech classification](#). In *Machine Learning and Knowledge Discovery in Databases. Applied Data Science and Demo Track*, volume 12461 of *Lecture Notes in Computer Science*, pages 423–439, Cham. Springer International Publishing.
- Anna Bavaresco, Raffaella Bernardi, Leonardo Bertolazzi, Desmond Elliott, Raquel Fernández, Albert Gatt, Esam Ghaleb, Mario Giulianelli, Michael Hanna, Alexander Koller, André F. T. Martins, Philipp Mondorf, Vera Neplenbroek, Sandro Pezzelle, Barbara Plank, David Schlangen, Alessandro Suglia, Aditya K Surikuchi, Ece Takmaz, and Alberto Testoni. 2025. [LLMs instead of human judges? a large scale empirical study across 20 NLP evaluation tasks](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*

- (Volume 2: Short Papers), pages 238–255, Vienna, Austria. Association for Computational Linguistics.
- Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2019. [Nuanced metrics for measuring unintended bias with real data for text classification](#). In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 491–500, New York, NY, USA. ACM.
- Maximin Coavoux, Shashi Narayan, and Shay B. Cohen. 2018. [Privacy-preserving neural representations of text](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1–10, Brussels, Belgium. Association for Computational Linguistics.
- Thomas Davidson, Dana Warmusley, Michael Macy, and Ingmar Weber. 2017. [Automated hate speech detection and the problem of offensive language](#). In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, pages 512–515.
- Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. [ERASER: A benchmark to evaluate rationalized NLP models](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online. Association for Computational Linguistics.
- Rotem Dror, Gili Baumer, Marina Bogomolov, and Roi Reichart. 2017. [Replicability analysis for natural language processing: Testing significance with multiple datasets](#). *Transactions of the Association for Computational Linguistics*, 5:471–486.
- Oluwaseyi Feyisetan, Borja Balle, Thomas Drake, and Tom Diethe. 2020. [Privacy- and utility-preserving textual analysis via calibrated multivariate perturbations](#). In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 178–186. ACM.
- Google Research. 2024. [google/civil\\_comments: Datasets at Hugging Face](#). Hugging Face dataset card. Accessed 2026-06-23.
- Laura Hanu and Unitary team. 2020. [Detoxify](#). Zenodo software record. <https://github.com/unitaryai/detoxify>.
- Hate-speech-CNERG. 2020. [Hate-speech-CNERG/dehatebert-mono-english: Hugging Face model card](#). Hugging Face model card. Accessed 2026-06-23.
- Sarthak Jain, Sarah Wiegrefe, Yuval Pinter, and Byron C. Wallace. 2020. [Learning to faithfully rationalize by construction](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4459–4473, Online. Association for Computational Linguistics.
- Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. [Rationalizing neural predictions](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117, Austin, Texas. Association for Computational Linguistics.
- Jiwei Li, Will Monroe, and Dan Jurafsky. 2016. [Understanding neural networks through representation erasure](#). *Preprint*, arXiv:1612.08220.
- Pierre Lison, Ildikó Pilán, David Sanchez, Montserrat Batet, and Lilja Øvrelid. 2021. [Anonymisation models for text data: State of the art, challenges and future directions](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4188–4203, Online. Association for Computational Linguistics.
- Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2021. [HateXplain: A benchmark dataset for explainable hate speech detection](#). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14867–14875.
- Justus Mattern, Benjamin Weggenmann, and Florian Kerschbaum. 2022. [The limits of word level differential privacy](#). In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 867–881, Seattle, United States. Association for Computational Linguistics.
- John Pavlopoulos, Jeffrey Sorensen, Léo Laugier, and Ion Androutsopoulos. 2021. [SemEval-2021 task 5: Toxic spans detection](#). In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 59–69. Association for Computational Linguistics.
- Ildikó Pilán, Pierre Lison, Lilja Øvrelid, Anthi Papadopoulou, David Sánchez, and Montserrat Batet. 2022. [The text anonymization benchmark \(TAB\): A dedicated corpus and evaluation framework for text anonymization](#). *Computational Linguistics*, 48(4):1053–1101.
- Yaxuan Ren, Krithika Ramesh, Yaxing Yao, and Anjalie Field. 2025. [How do we measure privacy in text? a survey of text anonymization metrics](#). In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 1532–1544, Mumbai, India. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics.
- Paul Röttger, Bertie Vidgen, Dong Nguyen, Zeerak Waseem, Helen Margetts, and Janet Pierrehumbert. 2021. [HateCheck: Functional tests for hate speech detection models](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1:*

*Long Papers*), pages 41–58, Online. Association for Computational Linguistics.

Samuel Sousa and Roman Kern. 2023. [How to keep text private? a systematic review of deep learning methods for privacy-preserving natural language processing](#). *Artificial Intelligence Review*, 56:1427–1492.

Toxic Spans Detection (SemEval-2021 Task 5). 2021. [heegyu/toxic-spans: Datasets at Hugging Face](#). Hugging Face dataset card. License CC0-1.0. Accessed 2026-06-24.

Zeera Waseem and Dirk Hovy. 2016. [Hateful symbols or hateful people? predictive features for hate speech detection on Twitter](#). In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California. Association for Computational Linguistics.

Oleksandr Yermilov, Vipul Raheja, and Artem Chernodub. 2023. [Privacy- and utility-preserving NLP with anonymized data: A case study of pseudonymization](#). In *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing*, pages 232–241, Toronto, Canada. Association for Computational Linguistics.

**Offensive Content Warning.** This appendix contains examples of hateful content and obfuscated slurs, included strictly to support this research. We have minimised the number of such examples and obfuscated all slurs. Readers who may find such content distressing should exercise caution.

## A Synthetic Release Examples

Following WOA’s guidance on reporting abusive content, these examples are synthetic and use placeholders rather than real social-media posts, slurs, or profanity. They are illustrative schematics, not sampled experiment rows. They show the context-recovery behavior only; the residual-identifier and classifier-transfer evaluations are reported in the main tables.

| Length                        | Synthetic source                                                                                                       | Importance only                             | Fixed window                                               | I-CEM                                                                                   |
|-------------------------------|------------------------------------------------------------------------------------------------------------------------|---------------------------------------------|------------------------------------------------------------|-----------------------------------------------------------------------------------------|
| Short                         | [PERSON] wrote: please do not call [GROUP] [SLUR].                                                                     | [SLUR]                                      | [GROUP] [SLUR]                                             | do not call [GROUP] [SLUR]                                                              |
| Medium                        | At the forum, [PERSON] quoted "[GROUP] are [SLUR]" and replied that the claim is hateful.                              | [GROUP] ... [SLUR]                          | quoted "[GROUP] are [SLUR]"                                | quoted "[GROUP] are [SLUR]" and replied that the claim is hateful                       |
| Multiple cues (label: normal) | [HANDLE] argued that [GROUP1] are not [SLUR] and [GROUP2] do not deserve [SLUR] either; the post defended both groups. | [GROUP1] ... [SLUR] ... [GROUP2] ... [SLUR] | [GROUP1] are not [SLUR] ... [GROUP2] do not deserve [SLUR] | [GROUP1] are not [SLUR] and [GROUP2] do not deserve [SLUR] either ... the post defended |

Table 4: Synthetic examples (all placeholders; no real posts or slurs). Rows 1–2 show the main failure mode: importance-only preserves the harmful cue but strips negation or quotation context; I-CEM recovers it via context rules. Row 3 shows the cost: when two target–harm pairs both fire (one per group), I-CEM expands to nearly the full sentence even on a non-hateful post defending those groups, while the fixed window stays more compressed.

## B Context Cue Inventory

The target and stance proxies match these fixed lexicons on lowercased tokens. Target cues also absorb any per-example target labels the dataset supplies, normalized to lowercase words. To follow WOA’s guidance on slur dictionaries, the harm-cue slurs are summarized and counted rather than reprinted; the verbatim list ships only in the released code.

- **Target** (identity-group references): arab(s), muslim(s), moslem(s), jew(s)/jewish, immigrant(s), refugee(s), woman/women, black, asian, gay, trans, christian(s), hindu(s), mexican(s), white(s), people; plus dataset-provided target labels.
- **Harm** (47 terms). About half are hostile-action or eliminationist words, reprinted here: ban(ned), deport(ed), leave, kick(ed), remove(d), out, kill(ed), threat(en), hate(d), inferior, dirty, filthy, vermin, shithole(s). The remaining 25 are derogatory slurs and hostile group labels targeting the groups above (anti-Muslim, antisemitic, anti-Black, homophobic, and ableist), obfuscated here (e.g.  $f * g$ ,  $k * * e$ ,  $r * * * * d$ ) and listed in full only in the code.
- **Negation**: not, never, no, cannot, can’t, don’t, doesn’t, isn’t, do, without.
- **Quotation/reporting**: said, says, say, saying, quote(d), called, claim(ed), wrote, posted, and straight/curly quotation marks.
- **Counterspeech**: condemn(ed), disagree, oppose(d), wrong, reject(ed), support, agree.

These inventories are deliberately small and English-only; they are a transparent baseline, not a complete cue lexicon, and their limited coverage is one source of the proxy error discussed in the Limitations. Several cues introduce known false-positive risk: do (negation list) is an auxiliary that only negates in combination with not, which is already covered by don’t/doesn’t; support and

agree (counterspeech list) fire on sentences endorsing rather than opposing harm (e.g. “I support banning [GROUP]”). More broadly, several cues are high-frequency function words (e.g. do, no, out, people), so cue survival partly reflects common-word retention; the independent stance judge (Appendix E.1) sidesteps this by reading meaning rather than matching tokens.

## C Multi-seed Variance

Table 5 reports the main HateXplain configuration rerun over four sampling seeds (17, 7, 23, 41), 5,000 rows each, with all other settings fixed. Values are mean  $\pm$  sample standard deviation across seeds. The I-CEM target and stance gains over the fixed window (about 22 and 37 points) are an order of magnitude larger than the across-seed standard deviation, while the positive-class F1 differences between variants stay within about 0.015. Because all four seeds draw 5,000 rows from the same HateXplain split under deterministic inference, this standard deviation reflects sampling variance over overlapping draws, not independent-dataset replicability (see Limitations).

| Variant           | F1+             | Flip          | Target        | Stance         |
|-------------------|-----------------|---------------|---------------|----------------|
| Importance only   | .704 $\pm$ .002 | 25.8 $\pm$ .6 | 54.8 $\pm$ .3 | 4.0 $\pm$ .3   |
| Imp. window (r=2) | .712 $\pm$ .005 | 17.5 $\pm$ .7 | 72.8 $\pm$ .8 | 24.4 $\pm$ .7  |
| I-CEM             | .698 $\pm$ .002 | 13.2 $\pm$ .4 | 95.1 $\pm$ .4 | 61.7 $\pm$ 1.0 |

Table 5: Multi-seed variance on HateXplain (seeds 17, 7, 23, 41; 5,000 rows each). Mean  $\pm$  std across seeds. Retained tokens are stable (I-CEM 49.9  $\pm$  0.1%, window 39.8  $\pm$  0.1%) and omitted for space. The context gaps far exceed seed variance.

## D Second-dataset Replication (Toxic Spans)

Table 6 reports SemEval-2021 Toxic Spans Detection (5,000 rows, seed 17, no PII injection, DeHateBERT). Rationale is overlap with the gold human toxic spans. F1+ is low because DeHateBERT is out of domain on this toxicity data, so this is a context/rationale replication, not a utility result. Two patterns transfer from HateXplain: (i) on raw human-flagged tokens I-CEM is comparable to a matched-budget window (rationale 79.2% vs. the radius-3 window’s 81.1% at a higher 54.3% token budget); (ii) I-CEM’s stance–harm advantage is large and survives the matched budget (64.2% vs. 30.1% for the radius-3 window), confirming the effect is not specific to HateXplain.

| Variant           | F1+  | Flip | Tok  | Stance | Rat. |
|-------------------|------|------|------|--------|------|
| Importance only   | .419 | 24.5 | 13.3 | 1.6    | 65.7 |
| Imp. window (r=2) | .347 | 11.7 | 45.5 | 22.5   | 77.6 |
| Imp. window (r=3) | .340 | 9.8  | 54.3 | 30.1   | 81.1 |
| I-CEM             | .341 | 10.6 | 50.7 | 64.2   | 79.2 |

Table 6: Toxic Spans Detection (SemEval-2021 Task 5, CC0; 5,000 rows, seed 17, no PII). Stance is stance–harm relation preservation; Rat. is overlap with gold human toxic spans. The radius-3 window is the token-matched baseline. I-CEM’s stance–*proxy* advantage replicates and holds at matched budget (this dataset was not independently judged); rationale overlap is comparable across I-CEM and matched windows.

## E Independent LLM Judge

Both the Stance and Target proxies are scored with the same cue lexicons I-CEM expands on, so we validate them with an independent, lexicon-free local judge that never sees the cue inventory. This appendix reports stance recovery (Appendix E.1) and label/target recovery (Appendix E.2).

### E.1 Stance recovery

An independent large language model (qwen3.5–9b, served locally; no text leaves the local network) labels the author’s stance toward the harmful content (ASSERT, NEGATE, QUOTE, OPPOSE, or UNCLEAR)

on the full text and on each release excerpt. The judge never sees I-CEM’s cue inventory. We take the 963 no-PII HateXplain rows (seed 17) where the lexical proxy fires, and let the judge define the denominator: the 228 rows whose *full text* reads as a non-assertion stance, i.e. where context genuinely changes the reading. A release preserves stance if its excerpt still reads as a non-assertion stance (Table 7).

### System prompt (stance judge).

You label the AUTHOR’S STANCE toward harmful or hateful content in a short text excerpt. The excerpt may be fragmentary (words joined by gaps). Pick exactly one:  
 ASSERT = the author states the harmful claim as their own view.  
 NEGATE = the harmful claim is explicitly denied, refused, or said to be false.  
 QUOTE = the harmful claim is explicitly quoted or attributed to another person, not endorsed.  
 OPPOSE = the author explicitly condemns, rejects, or argues against the harmful claim.  
 UNCLEAR = too fragmentary or missing context to tell the stance.  
 Use NEGATE/QUOTE/OPPOSE only when the excerpt itself shows that signal. If the excerpt is just a bare harmful phrase with no negation/attribution/opposition visible, label ASSERT (or UNCLEAR if it is only disconnected words).  
 Reply with ONLY the single label word.

The excerpt is passed as the user turn verbatim, with no additional framing. For example, the importance-only release [GROUP] . . . [SLUR] is passed as-is and typically returns ASSERT or UNCLEAR; the I-CEM release [GROUP] [SLUR] is wrong . . . do not hold that view typically returns NEGATE or OPPOSE. Temperature is 0; max\_tokens is 6.

| Variant                     | Tok  | Stance preserved (95% CI) |
|-----------------------------|------|---------------------------|
| Importance only             | 14.3 | 1.3 [0.0, 3.1]            |
| Fixed window (r=2)          | 49.7 | 26.3 [20.6, 32.0]         |
| Fixed window (r=3, matched) | 59.7 | 42.1 [36.0, 48.7]         |
| I-CEM                       | 57.5 | 49.1 [42.5, 55.7]         |

Table 7: Independent LLM stance judge on 228 HateXplain rows whose full text reads as a non-assertion stance. Stance preserved is the fraction whose release excerpt still reads as a non-assertion stance to the judge; CIs are 10k-sample bootstrap. Every reduction beats importance-only with McNemar  $p < 10^{-4}$ . I-CEM exceeds the token-matched r=3 window (49.1 vs. 42.1) but the paired difference is not significant (McNemar  $p = 0.052$ ), so the lexical proxy’s  $1.8\times$  matched-budget gap overstates the grounded effect; the robust result is the collapse of stance under importance-only and its recovery under any context expansion.

Because the single matched-point difference is not significant, we describe the exposure–stance frontier rather than test it. Judging the same 228 rows across window radii traces it: r=0 (importance-only) 14.3% tokens / 1.3% stance, r=1 35.5% / 11.0%, r=2 49.7% / 26.3%, r=3 59.7% / 42.1%. I-CEM sits at 57.5% / 49.1%, 10.5 points of stance above the window frontier interpolated at its own token budget (38.6%). This is a descriptive comparison across operating points, not a hypothesis test: the only measured window near I-CEM’s budget is r=3, and that is the non-significant paired difference above.

## E.2 Label and target recovery

The same local judge (qwen3.5–9b, LAN-only) measures whether a release preserves the *label* a reader would assign, on all 963 context-sensitive no-PII HateXplain rows (seed 17). Given only an excerpt, it answers (i) is this hateful/offensive toward a group? and (ii) which group is targeted? We compare each release to the judge’s reading of the full text. *HateFid* is agreement of the excerpt’s hate decision with the full-text decision; *CS→Hate* is the fraction of the 200 full-text-non-hateful rows (counter-speech, negation, quotation) that the excerpt flips to hateful; *TgtRec* is target recovery on the 670 rows with an identifiable full-text target (Table 8).

**System prompts (label recovery judge).** Each excerpt is passed to two sequential calls; both receive the excerpt as the user turn verbatim.

*Hate decision:*

You judge whether a short text excerpt is hateful or offensive toward a group of people. The excerpt may be fragmentary (words joined by gaps or [MASKED] placeholders). Answer with exactly one word:  
YES = the excerpt expresses or endorses hate/offence toward a group.  
NO = the excerpt does not (including when it denies, quotes-without-endorsing, or opposes such content).  
UNCLEAR = too fragmentary to tell.  
Reply with ONLY YES, NO, or UNCLEAR.

*Target identification:*

You identify which group of people a short text excerpt targets or is about. The excerpt may be fragmentary. Reply with the group in 1-3 lowercase words (e.g. 'muslims', 'gay people', 'women', 'black people', 'immigrants', 'jews'). If no group is identifiable, reply NONE. Reply with ONLY the group or NONE.

For example, the importance-only release [GROUP] ... [SLUR] typically returns YES / [GROUP] (bare harmful phrase reads as assertion); the I-CEM release [GROUP] [SLUR] is wrong ... do not hold that view typically returns NO / [GROUP]. Temperature is 0; max\_tokens is 8 for hate, 12 for target. In multi-judge runs (ICEM\_JUDGE\_TOOLS=1) the same prompts are delivered via a structured tool call to support reasoning-tuned models that emit their answer in reasoning\_content rather than content.

| Variant         | HateFid (CI)     | CS→Hate | TgtRec (CI)      |
|-----------------|------------------|---------|------------------|
| Importance only | 76.9 [74.2,79.6] | 53.0    | 35.5 [31.9,39.1] |
| Window (r=2)    | 84.9 [82.6,87.1] | 34.5    | 52.1 [48.4,55.8] |
| Window (r=3)    | 87.7 [85.7,89.8] | 28.0    | 58.2 [54.5,61.9] |
| I-CEM           | 89.7 [87.7,91.7] | 24.5    | 67.3 [63.7,70.7] |

Table 8: Independent label/target recovery (963 rows; CIs are 10k-sample bootstrap). Lower CS→Hate is better. I-CEM vs. the token-matched r=3 window: HateFid McNemar  $p = 0.04$ , TgtRec  $p < 10^{-4}$ ; both vs. importance-only  $p < 10^{-4}$ . Naive importance-only release misreads over half of non-hateful content as hateful and loses two thirds of targets; I-CEM significantly limits both.

**Grounding against gold labels.** Because the reference above is the judge’s own full-text reading, we re-anchor the two label-encoding metrics to the datasets’ gold annotations, which need no new human labelling (Table 9; scripts/analyze\_label\_recovery\_vs\_gold.py). First, the judge is a usable instrument: its full-text hate decision matches the gold binary label on 80.0% of the 963 rows (recall 91.7%, precision 82.2%). Second, target recovery against the gold HateXplain target groups (using a fixed synonym map that aligns the judge’s natural-language groups with HateXplain’s controlled vocabulary, e.g. “black people”≡african, “muslims”≡islam) preserves the ordering and the significant I-CEM advantage: importance-only 36.7%, window r=2 46.9%, token-matched window r=3 50.3%, I-CEM 56.0% (McNemar I-CEM vs. matched window  $p=.0003$ ; vs. importance-only  $p < 10^{-4}$ ). Hate-label accuracy against gold rises monotonically (importance-only 73.2%, matched window 76.6%, I-CEM 78.1%) but the I-CEM-vs-matched gap is not significant ( $p=.11$ ), matching its borderline significance against the judge. The counter-speech misread metric is deliberately *not* re-anchored to gold: HateXplain labels do not encode stance (counter-speech quoting a slur is usually labelled hateful), so against gold non-hate rows every variant misreads 50–60% and the contrast collapses; the judge’s stance-aware full-text read is the appropriate reference for that metric.

**Robustness across judges.** To check that the recovery result is not an artifact of one judge, we replicate it with two further independent local judges from different vendors and scales (gemma-4-26b, Google; gpt-oss-20b, OpenAI), plus qwen3.5-9b re-run under a tool-calling protocol as a control.

| Variant               | TgtRec      |             | Hate label  |             |
|-----------------------|-------------|-------------|-------------|-------------|
|                       | judge       | gold        | judge       | gold        |
| Importance only       | 35.5        | 36.7        | 76.9        | 73.2        |
| Window (r=2)          | 52.1        | 46.9        | 84.9        | 75.1        |
| Window (r=3, matched) | 58.2        | 50.3        | 87.7        | 76.6        |
| I-CEM                 | <b>67.3</b> | <b>56.0</b> | <b>89.7</b> | <b>78.1</b> |

Table 9: Recovery against the judge’s full-text read vs. the datasets’ gold labels (no new annotation). *TgtRec* is target recovery (judge: vs. judge full-text,  $n=670$ ; gold: vs. gold target groups with a synonym map,  $n=750$ ). *Hate label* is excerpt-vs-full-text agreement (judge,  $n=963$ ) and excerpt-vs-gold accuracy (gold,  $n=951$ ). I-CEM vs. the token-matched window: TgtRec significant under both references (judge  $p < 10^{-4}$ , gold  $p=.0003$ ); the hate-label edge is significant against the judge ( $p=.04$ ) but not against gold ( $p=.11$ ), so we report it as directional. The judge’s own full-text hate read agrees with the gold label 80.0% of the time.

Reasoning-tuned judges only return a usable short answer when forced to emit a structured tool call; we run the panel on a deterministic 400-row subsample (LM Studio serialises requests, so the full set is prohibitively slow). The ordering is unanimous: every judge ranks I-CEM  $\geq$  the token-matched window and importance-only worst on both HateFid and TgtRec (Table 10). Target recovery favours I-CEM significantly for three of four judges; the label-fidelity margin over the *wide* matched window is significant for both Qwen judges and directional (n.s. at  $n=400$ ) for the other two, consistent with the borderline matched-budget gap in the main run. The counter-speech misread rate behaves the same way: every judge places I-CEM well below importance-only, but against the matched window I-CEM is lower for both Qwen judges, tied for gemma-26b, and slightly higher for gpt-oss, consistent with the non-significant  $p=0.31$  margin in the main run. The robust, judge-independent finding is the corruption itself and the importance-only  $\rightarrow$  context-recovery ordering, not a sharp I-CEM-vs-wide-window gap.

| Judge ( $n$ )                                        | ImpOnly | Matched | I-CEM | $p$       |
|------------------------------------------------------|---------|---------|-------|-----------|
| <i>HateFid (excerpt vs. full-text hate decision)</i> |         |         |       |           |
| qwen plain (963)                                     | 76.9    | 87.7    | 89.7  | 0.04*     |
| qwen tool (400)                                      | 61.2    | 77.8    | 83.0  | 0.01*     |
| gemma-26b (400)                                      | 53.1    | 79.6    | 82.1  | 0.25      |
| gpt-oss (400)                                        | 65.7    | 81.9    | 84.4  | 0.24      |
| <i>TgtRec (target recovery)</i>                      |         |         |       |           |
| qwen plain (670)                                     | 35.5    | 58.2    | 67.3  | $<1e-4^*$ |
| qwen tool (400)                                      | 50.3    | 67.5    | 73.2  | 0.02*     |
| gemma-26b (400)                                      | 58.9    | 73.4    | 75.3  | 0.43      |
| gpt-oss (400)                                        | 56.6    | 69.5    | 75.9  | 0.02*     |

Table 10: Multi-judge replication of label/target recovery. ImpOnly = importance-only, Matched = token-matched  $r=3$  window.  $p$  = McNemar exact, I-CEM vs. matched window; \* significant at 0.05.  $n$  in parentheses is the hate-fidelity denominator (rows judged); TgtRec denominators are smaller (rows with an identifiable full-text target: 670 for qwen plain, 314/316/348 for the 400-row panel). Direction is unanimous across judges; significance of the I-CEM-vs-wide-window margin is judge-dependent.