CERTIFYING GENERATIVE LLMS AGAINST SUFFIX-BASED JAILBREAKS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose **Discrete-Continuous Randomized Smoothing (DCRS)**, to our knowledge the first framework that certifies the safety of smoothed LLM responses against suffix-based jailbreaks while accounting for both discrete token changes and continuous embedding displacement. Existing certified defenses adapt ℓ_0-style (masking- or erasure-based) certificates from text classification to the safety setting: they bound the number of altered tokens but ignore how far those tokens move in embedding space, and extensive masking can discard useful input information. DCRS instead combines discrete token selection with Gaussian embedding perturbations and applies hierarchical randomized smoothing to derive an ℓ_2 certified radius ϵ_k for changes to at most k token positions. The smoothed decision remains SAFE when at most k token positions change and their joint ℓ_2 embedding displacement is below ϵ_k. We specialize this guarantee to suffix attacks by padding the reference prompt with dummy tokens and representing suffix insertion as fixed-length replacement. This construction yields a sufficient condition for certifying every vocabulary-token assignment to the suffix slots, connecting continuous embedding radii to discrete adversarial suffixes. Experiments demonstrate non-vacuous certificates for multi-token suffix replacements and evaluate empirical jailbreak resistance alongside benign-task utility.

1 INTRODUCTION

Suffix-based jailbreaks form a major family of attacks on safety-aligned large language models (LLMs): an adversary appends a token sequence to a harmful request to induce an unsafe response. GCG established that optimized suffixes can transfer across requests and models (Zou et al., 2023). Subsequent work has developed complementary ways to construct them, including gradient-free beam search in BEAST (Sadasivan et al., 2024), adaptive random search using output probabilities (Andriushchenko et al., 2025), and learned suffix generation in AmpleGCG (Liao & Sun, 2024). Suffix optimization also features in standardized evaluations such as HarmBench and JailbreakBench (Mazeika et al., 2024; Chao et al., 2024). The diversity of suffix-generation methods motivates guarantees that hold across all suffixes within a specified threat model, independently of the algorithm used to construct them.

Empirical evaluations measure a defense's resistance to attacks produced by particular algorithms, but leave open whether other suffixes can still induce unsafe responses (Andriushchenko et al., 2025; Nasr et al., 2025). This gap motivates robustness certification for generative LLMs, where the property to be guaranteed is the safety of the generated response. However, existing certified defenses operate at the level of a text classifier or a harmful-prompt filter rather than the generated response (Ye et al., 2020; Zhao et al., 2022; Zeng et al., 2023; Huang et al., 2023; Ji et al., 2024; Huang et al., 2024; Lou et al., 2024; Wang et al., 2026; Kumar et al., 2024). Our goal is to connect discrete changes in the input to a provable guarantee on this response-level safety decision.

The standard tool for such guarantees is randomized smoothing, which constructs a prediction rule by aggregating predictions under random input perturbations and uses the resulting class probabilities to certify stability against adversarial changes (Lécuyer et al., 2019; Cohen et al., 2019). In NLP, randomized smoothing has mainly been applied to text classification under discrete perturbation budgets, including ℓ_0-style bounds on word changes and bounds on edit distance (Ye et al., 2020; Zeng et al., 2023; Ji et al., 2024; Huang et al., 2024). An ℓ_0 budget counts altered positions without

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

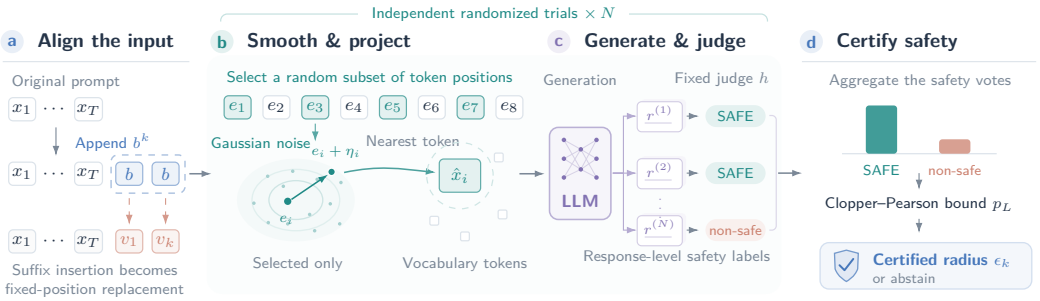


Figure 1: Overview of DCRS. (a) Dummy padding represents suffix insertion as fixed-position replacement. (b) Each trial selects token positions, adds Gaussian embedding noise, and projects to vocabulary tokens. (c) The LLM generates a response, which a fixed judge labels. (d) Aggregated safety labels yield a probability lower bound and a certified radius, or abstention.

distinguishing the magnitude of the corresponding embedding displacement. Moreover, masking-based smoothing can discard useful input information as the masking rate increases, motivating denoising to recover the missing context (Ji et al., 2024). These limitations motivate incorporating continuous embedding geometry into discrete text certification, so that robustness depends on both how many tokens change and how far their embeddings move.

We introduce **Discrete-Continuous Randomized Smoothing (DCRS)**, a framework that connects discrete token edits and continuous embedding changes to the safety of generated responses. As illustrated in Figure 1, DCRS selects a subset of token positions and adds Gaussian noise to their embeddings before vocabulary projection and response generation. We formulate projection, generation, and response judging as a single decision rule, allowing input perturbations to be analyzed through their effect on the final safety label. For changes to at most k token positions, we derive an ℓ_2 certified radius ϵ_k within which the smoothed safety decision remains safe. This formulation brings the joint perturbation model of hierarchical randomized smoothing (Scholten et al., 2023) into a generative jailbreak setting, with the certificate defined on judged responses. The method operates on a pretrained generative model without retraining or an additional LLM denoising stage.

We further develop a construction that turns this response-level certificate into a guarantee against adversarial suffixes. Appending dummy tokens to the reference prompt represents suffix insertion as replacement of a fixed set of positions. We then bound the largest embedding change attainable by any vocabulary-token assignment to these positions. When the certified radius exceeds this bound, every suffix of the corresponding length is covered under the padded-input construction. Applying the condition at each length up to a chosen budget covers every suffix of at most that length, independently of how an attacker constructs the suffix. To our knowledge, DCRS is the first framework that certifies the safety of smoothed LLM responses against suffix-based jailbreaks while accounting for both discrete token changes and continuous embedding displacement.

Our contributions are:

- **Discrete-continuous response certification.** We connect token changes and embedding displacement to generated-response safety by incorporating projection, generation, and judging into one certified decision rule. The resulting radius accounts for both how many tokens change and how far their embeddings move.
- **Certification against arbitrary bounded-length suffixes.** We express suffix insertion as replacement of padded positions and derive a condition covering every vocabulary-token assignment. Satisfying this condition at each length certifies every suffix of at most a chosen length, independently of the attack algorithm.
- **Evaluation against practical defense desiderata.** Following the desiderata of Robey et al. (2025), we systematically evaluate DCRS in terms of attack mitigation, benign-task utility, computational cost, and compatibility with open-weight generative LLMs. Our experiments demonstrate non-vacuous suffix certificates alongside reduced empirical attack success rates and retained benign utility, and quantify the generation cost of obtaining these guarantees.

2 Related Work

Jailbreak attacks and empirical defenses. Beyond suffix attacks, jailbreaks arise from handcrafted prompts and from automated search such as AutoDAN, TAP, and PAIR (Wei et al., 2023; Liu et al., 2024; Mehrotra et al., 2024; Chao et al., 2025). Defenses include perplexity filtering (Alon & Kamfonas, 2023), guardrail classifiers (Li et al., 2025), representation-level training (Zou et al., 2024), and defensive prompt optimization such as RPO and PAT (Zhou et al., 2024; Mo et al., 2024). Adaptive adversaries that account for the deployed mechanism recover high success rates (Andriushchenko et al., 2025; Nasr et al., 2025), so empirical evaluation measures resistance to specific algorithms rather than to a perturbation set.

Certified robustness for text classification. Randomized smoothing for text has mainly certified classifiers under discrete budgets: SAFER randomizes synonym substitutions (Ye et al., 2020), RanMASK masks words to bound the number of changes (Zeng et al., 2023), and RS-Del and CERT-ED use randomized deletion for edit distance (Huang et al., 2023; 2024). These certificates treat every substitution within a budget identically, regardless of how far the replaced token moves in embedding space. Masking also discards input information, so SelfDenoise and CluCERT reconstruct the masked text with an additional LLM pass (Ji et al., 2024; Wang et al., 2026), at the cost of a second generation per trial. In all cases the certificate concerns a classifier label rather than a generated response.

Certified jailbreak defenses. Existing certificates for jailbreaks have mainly targeted input filters: erase-and-check certifies a harmful-prompt detector (Kumar et al., 2024), and Chen et al. (2025) bound the worst-case robustness of a safety detector under general smoothing distributions. These guarantees do not constrain what the model generates once a prompt passes. SmoothLLM aggregates responses over character-perturbed prompts (Robey et al., 2025), but its guarantee assumes that adversarial suffixes are k-unstable (Kumarappan & Mehrotra, 2025) and thus depends on a property of the attack rather than on the smoothing distribution. DCRS instead certifies the smoothed decision of a rule containing projection, generation, and a fixed judge, over an explicit perturbation set and without assumptions on how the attack is constructed.

Hierarchical randomized smoothing. Gaussian smoothing gives ℓ_2 certificates (Lécuyer et al., 2019; Cohen et al., 2019), randomized ablation gives ℓ_0 certificates (Levine & Feizi, 2020), and hierarchical randomized smoothing (HRS) combines entity selection with continuous perturbation of the selected entities (Scholten et al., 2023). DCRS specializes HRS to token positions and a response-level safety decision, and adds the padding construction that converts the continuous radius into coverage of every token assignment to the suffix slots.

3 Method

DCRS combines discrete selection of token positions with continuous perturbations of their embeddings. We first define the randomized generation procedure and its response-level safety decision, then describe how padding represents suffix insertion as token replacement. Section 4 establishes the corresponding robustness guarantees.

3.1 Smoothing and response-level safety decisions

Let an input prompt be a token sequence $x = (x_1, \ldots, x_T)$ of length T, where each $x_i \in V$ is a token from the vocabulary. Let $E : V \to \mathbb{R}^d$ denote the token embedding lookup, and write $e = (e_1, \ldots, e_T)$ with $e_i = E(x_i) \in \mathbb{R}^d$ for the corresponding embedding sequence. A generative model f maps an embedding sequence to a response in $\mathcal{R}$, and a deterministic judge $h : \mathcal{R} \to \mathcal{Y}$ assigns a response-level label. We fix the tokenizer, embedding map, decoding configuration, and judge throughout certification.

We define the smoothing distribution on the embedding sequence e. In each randomized trial, we first sample a subset of token positions

$$S \sim \mathrm{Unif}\{S \subseteq [T] : |S| = m\}, \tag{1}$$

where m is the number of selected positions. Given S, we define the embedding-space perturbation operator ϕ_S by

$$\phi_S(e_i) = \begin{cases} e_i + \eta_i, & i \in S, \\ e_i, & i \notin S, \end{cases} \qquad \eta_i \sim \mathcal{N}(0, \sigma^2 I_d). \tag{2}$$

Thus, ϕ_S adds Gaussian noise only to the selected token embeddings. The subset sampling acts as a **discrete** upper-level smoothing step over token positions, while ϕ_S provides a **continuous** lower-level smoothing step in embedding space.

Since a noisy embedding vector may not correspond to a valid vocabulary embedding, we apply a deterministic nearest-neighbor projection under cosine similarity before generation. Let

$$\mathrm{NN}_V(z) = \arg\max_{v \in V} \frac{\langle z, E(v) \rangle}{\|z\|_2 \, \|E(v)\|_2} \tag{3}$$

denote the vocabulary token with the highest cosine similarity to a vector $z \in \mathbb{R}^d$. We define the token-wise projection-and-embedding map $D : \mathbb{R}^d \to \mathbb{R}^d$ as

$$D(z) = E(\mathrm{NN}_V(z)) \,. \tag{4}$$

That is, D first projects each noisy embedding to its nearest vocabulary token and then maps the projected token back to its embedding. This step ensures that the perturbed sequence passed to the generative model corresponds to valid vocabulary embeddings, i.e., $D(\mathbb{R}^d) \subseteq E(V)$.

The deterministic base decision rule being smoothed is the composite map $H : (\mathbb{R}^d)^T \to \mathcal{Y}$,

$$H(z) = h(f(D(z))) \,. \tag{5}$$

H includes nearest-neighbor projection, generation, and response-level decision mapping. The smoothed decision is then

$$g(e) = \arg\max_{y \in \mathcal{Y}} \Pr_{S,\eta} \left(H(\phi_S(e)) = y \right). \tag{6}$$

Gaussian randomized smoothing applies to arbitrary deterministic base rules (Lécuyer et al., 2019; Cohen et al., 2019), including discontinuous vocabulary projection. This treatment follows the composite-classifier construction used in denoised smoothing (Salman et al., 2020; Carlini et al., 2023; Jeong & Shin, 2023).

For certification, we use $\mathcal{Y} = \{\textsc{Safe}, \textsc{Not-Safe}\}$. Responses that match the refusal-prefix rule of Appendix C.4 count toward the SAFE class; all other responses, including empty generations, count as NOT-SAFE. This conservative assignment ensures that only explicit refusals contribute to the safety probability used for certification. From N independent trials and n_{safe} safe outcomes, we compute a one-sided Clopper–Pearson lower confidence bound p_L on the safe-response probability (Clopper & Pearson, 1934). The procedure returns a safety certificate when this bound satisfies the conditions in Section 4 and abstains otherwise.

3.2 Input alignment for suffix attacks

Suffix attacks append adversarial tokens to a prompt (Zou et al., 2023; Sadasivan et al., 2024; Liao & Sun, 2024; Jia et al., 2025). To compare the reference prompt and the attacked input in a common position space, we append dummy tokens to the reference. For a suffix length k, define

$$\bar{x}^{(k)} = x \oplus b^k, \tag{7}$$

where $\oplus$ denotes concatenation, b^k consists of k copies of a fixed dummy token b, and $T_k := |\bar{x}^{(k)}|$. An adversarial suffix $s \in V^k$ produces $x \oplus s$ by replacing these k placeholders. The dummy token therefore supplies a reference embedding for each suffix position. It is not intended to approximate any adversarial suffix; its role is to place the reference prompt and every suffix-augmented prompt of the same length in a common fixed-length certification space.

We apply the smoothing procedure of Section 3.1 to the entire padded sequence, using a subset of m_k positions sampled uniformly from its T_k positions. We do not condition the sampled subset on the suffix slots, so the certificate holds under the conservative threat model in which any set of at most k positions of the padded input may change, independently of how the suffix is constructed. Each padded length is evaluated separately, with its own safe-response probability lower bound. This corresponds to step (a) in Figure 1; Section 4.2 establishes when the resulting radius covers every replacement of the suffix slots.

4 THEORETICAL GUARANTEES

We analyze the smoothed response decision defined in Section 3.1. We first derive a certificate under a joint constraint on the number of modified token positions and their embedding displacement. We then translate this guarantee into coverage of arbitrary vocabulary assignments to padded suffix slots.

4.1 CERTIFICATION UNDER TOKEN-LEVEL AND EMBEDDING-SPACE PERTURBATIONS

We specialize the HRS reduction of Scholten et al. (2023) to token positions with Gaussian noise on the selected embedding blocks.

Let $y_A = g(e)$ be the smoothed decision of the clean input, and let p_L be a lower confidence bound on

$$\Pr_{S,\eta}\left(H(\phi_S(e)) = y_A\right),$$

where S is sampled as in Equation (1) and ϕ_S is the embedding-space perturbation operator defined in Equation (2). We consider an adversarial embedding sequence e' that differs from e only on a set of token positions $C \subseteq [T]$. We write e_C and e'_C for the embedding blocks restricted to C, and measure the embedding-space displacement by $\|e'_C - e_C\|_2$.

The following reduction separates subset coverage from robustness within the selected block.

Theorem 4.1 (HRS reduction, restated from Scholten et al. (2023))**.** *Let C be the set of perturbed entities, and let Δ denote the probability of not selecting all entities in C in the upper-level smoothing step. Suppose the clean input has label probability lower bounded by p_L and $p_L > \Delta$. Define*

$$p'_L = \frac{p_L - \Delta}{1 - \Delta}. \tag{8}$$

If the lower-level smoothing distribution gives a probability lower bound $q_\epsilon(p'_L)$ under a perturbation of size ϵ on the selected entities, then the hierarchical smoothed probability of the same label on the perturbed input is lower bounded by

$$(1 - \Delta) q_\epsilon(p'_L). \tag{9}$$

In our setting the entities are token positions, and the event of selecting every modified position has probability $1-\Delta$. Conditional on this event, the lower-level certificate controls the embedding change. Appendix A gives the corresponding Neyman–Pearson derivation (Neyman & Pearson, 1933). For the fixed-size sampler in Equation (1), the coverage probability has the following combinatorial form.

Theorem 4.2 (Fixed-size token subset certificate)**.** *Assume that each randomized trial selects exactly m token positions uniformly at random, as in Equation* (1)*. Let an adversarial embedding sequence e' differ from e only on a set $C \subseteq [T]$ with $|C| = k$, and suppose*

$$\|e'_C - e_C\|_2 \le \epsilon. \tag{10}$$

For $k \le m \le T$, define

$$\Delta_k = 1 - \frac{\binom{T-k}{m-k}}{\binom{T}{m}}. \tag{11}$$

If $p_L > \Delta_k$, then the probability that the randomized procedure returns the clean smoothed decision y_A on the adversarial input is lower bounded by

$$p_{\mathrm{cert}}(k, \epsilon) = (1 - \Delta_k)\Phi\left(\Phi^{-1}\left(\frac{p_L - \Delta_k}{1 - \Delta_k}\right) - \frac{\epsilon}{\sigma}\right), \tag{12}$$

where Φ is the CDF of the standard normal distribution. If $k > m$, the certificate is vacuous.

Proof. The total number of token subsets of size m is $\binom{T}{m}$. A sampled subset S contains all perturbed positions in C if and only if the remaining $m - k$ positions are chosen from the $T - k$ positions outside C. Therefore,

$$\Pr(C \subseteq S) = \frac{\binom{T-k}{m-k}}{\binom{T}{m}}. \tag{13}$$

Thus the probability of not selecting all perturbed positions is $\Delta_k = 1 - \Pr(C \subseteq S)$, which gives Equation (11).

Substituting $\Delta = \Delta_k$ into Theorem 4.1 gives $p'_L = (p_L - \Delta_k)/(1 - \Delta_k)$. On the event $C \subseteq S$, the clean and adversarial selected embedding blocks differ only on the coordinates in C, so their total selected-block displacement is $\|e'_C - e_C\|_2$. The lower-level smoothing distribution on the selected embeddings is isotropic Gaussian noise with variance $\sigma^2 I_d$. The standard Gaussian randomized smoothing bound for H (Cohen et al., 2019) gives, for every perturbation satisfying Equation (10),

$$q_\epsilon(p'_L) = \Phi\left(\Phi^{-1}(p'_L) - \frac{\epsilon}{\sigma}\right).$$

Multiplying this lower-level bound by the probability of selecting all perturbed positions, $1 - \Delta_k$, yields Equation (12). □

Corollary 4.3 (Binary safety certificate)**.** *For a fixed token budget k, consider all perturbations with $|C| \le k$ and $\|e'_C - e_C\|_2 \le \epsilon$. Since Δ_j is nondecreasing in j for fixed T and m, Δ_k upper bounds the combinatorial penalty for all sets of size at most k. Then, in the binary safety setting, the smoothed decision is certified whenever*

$$p_{\mathrm{cert}}(k, \epsilon) > \frac{1}{2}. \tag{14}$$

When $p_L > \frac{1}{2} + \Delta_k$, this condition is equivalent to

$$\epsilon < \sigma\left[\Phi^{-1}\left(\frac{p_L - \Delta_k}{1 - \Delta_k}\right) - \Phi^{-1}\left(\frac{1}{2(1 - \Delta_k)}\right)\right]. \tag{15}$$

If $p_L \le \frac{1}{2} + \Delta_k$, the certificate does not yield a positive binary certified radius for budget k. The proof follows by applying Theorem 4.2, using the monotonicity of Δ_k in the token budget, and solving the condition $p_{\mathrm{cert}}(k, \epsilon) > 1/2$ for ϵ; we provide the details in Appendix B.

Sampling trade-off. Increasing $r = m/T$ improves subset coverage and lowers Δ_k. It also changes the distribution of the remaining embeddings and can reduce the safe-response bound p_L. Equation (15) expresses the resulting trade-off: the radius depends on both coverage and response stability. Appendices E and F examine this dependence, including the full-sequence endpoint $m = T$.

4.2 From embedding radii to suffix guarantees

The padding construction in Section 3.2 represents a suffix attack as changes to the final k positions of $\bar{x}^{(k)}$. The certificate in Section 4.1 covers a larger set of perturbations that may change any k positions of this fixed-length input:

$$\mathcal{A}_{k,\epsilon}(\bar{x}^{(k)}) = \left\{x' : |C| \le k,\ \left\|E(x')_C - E(\bar{x}^{(k)})_C\right\|_2 \le \epsilon\right\}. \tag{16}$$

Here x' has length T_k, and C is the set of positions at which x' and $\bar{x}^{(k)}$ differ. For a modified set of size k, the probability that all modified positions are selected is $\binom{T_k - k}{m_k - k}/\binom{T_k}{m_k}$. Thus, the padded-input penalty is

$$\Delta_k = 1 - \frac{\binom{T_k - k}{m_k - k}}{\binom{T_k}{m_k}}. \tag{17}$$

Substituting this penalty and the padded-input safe-probability bound into Equation (15) yields the binary certified radius ϵ_k.

To connect this radius to arbitrary vocabulary tokens, define the maximum displacement from the dummy embedding as

$$\rho = \max_{v \in V} \|E(v) - E(b)\|_2. \tag{18}$$

This quantity depends on the placeholder and embedding table, and bounds the change at each suffix position.

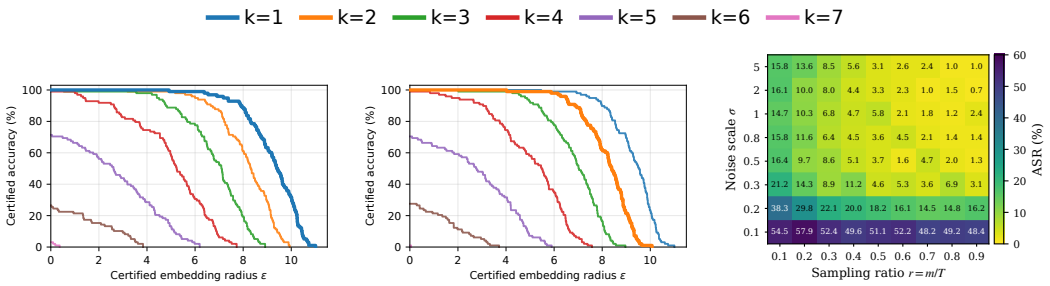


Figure 2: Certified accuracy versus embedding radius for one (left) and two (middle) dummy suffix tokens, and empirical ASR of DCRS over noise scale and sampling ratio (right). Certification curves use all 98 prompts under the main configuration of Section 5.1 ($\sigma = 5$, $r = 0.9$). Each curve fixes the perturbation budget k and varies the embedding radius ϵ; bold curves use a budget equal to the number of suffix slots.

Corollary 4.4 (Vocabulary-wide suffix certificate)**.** *Let ϵ_k be a positive binary safety radius for the padded reference $\bar{x}^{(k)} = x \oplus b^k$, as given by Corollary 4.3. If*

$$\epsilon_k > \rho\sqrt{k}, \tag{19}$$

then $g(E(x \oplus s)) =$ SAFE for every suffix $s \in V^k$ under the same padded-input smoothing configuration.

Proof. The sequences $x \oplus s$ and $x \oplus b^k$ differ in at most k positions. Writing $s = (s_1, \ldots, s_k)$, their joint embedding displacement satisfies

$$\left\|E(x \oplus s) - E(x \oplus b^k)\right\|_2 = \left(\sum_{i=1}^{k} \|E(s_i) - E(b)\|_2^2\right)^{1/2} \leq \rho\sqrt{k} < \epsilon_k.$$

Every such suffix therefore lies strictly within the binary safety radius, so Corollary 4.3 preserves the safe decision. □

Coverage through suffix length K requires Equation (19) to hold separately for every padded input $x \oplus b^k$, $k = 1, \ldots, K$, using its own length T_k, subset size m_k, and safe-probability bound. This gives a guarantee over all vocabulary assignments at each covered length, independently of the algorithm used to construct an attack. Appendix L gives the corresponding certification procedure.

5 EXPERIMENTS

We evaluate certified suffix robustness, attack success rate (ASR), and benign-task accuracy. Certified accuracy measures the fraction of prompts satisfying the vocabulary-wide replacement criterion. ASR measures success on the evaluated attack instances, including attacks beyond the certified perturbation set.

5.1 EXPERIMENTAL SETUP

For certification, we use Llama-2-7B-Chat on 98 AdvBench harmful requests (Touvron et al., 2023; Zou et al., 2023), with $N = 2{,}000$ trials, a one-sided 95% Clopper–Pearson bound, $\sigma = 5$, cosine nearest-neighbor projection, and sampling ratio $r = 0.9$, implemented as $m_k = \max\{1, \text{round}(0.9T_k)\}$ so that the effective ratio varies with input length. The placeholder is Llama token ! (ID 1738) with $\rho = 1.5697$, and each input is smoothed separately with $k = 1, \ldots, 5$ placeholders. The judge h is a fixed refusal-prefix rule (Appendix C.4).

For empirical robustness, we evaluate Llama-2-7B-Chat, Vicuna-7B-v1.3 (Chiang et al., 2023), and gpt-oss-20b (OpenAI, 2025b), and compare DCRS with SelfDenoise, SmoothLLM, RPO, PAT, perplexity filtering (PPL), Circuit Breakers (CB), PIGuard, and an undefended model (Ji et al., 2024; Robey et al., 2025; Zhou et al., 2024; Mo et al., 2024; Alon & Kamfonas, 2023; Zou et al., 2024; Li

Table 1: Certified accuracy (↑) and empirical attack success rate (ASR, ↓), in %. The Llama-2 certification row covers one-token suffix replacements at $\sigma = 1$, whereas GCG uses 20-token suffixes. Parenthesized values for PPL and PIGuard indicate undetected jailbreak rates.

Model	Attack / Metric	DCRS	SelfDenoise	SmoothLLM	RPO	PAT	PPL	CB	PIGuard	None
Llama-2	$k = 1$ certified ↑	**98.98**	**46**				***Non-certifiable defenses***			
	GCG ($k = 20$) ↓	1.0	4	1	9	2	0(0)	20	11(28)	55
	PAIR ↓	4	4	0	4	0	86(100)	2	60(68)	86
	AutoDAN ↓	0	0	0	0	0	3(100)	0	0(33)	3
Vicuna	GCG ($k = 20$) ↓	4.8	14	12	7.3	2.4	0(0)	12	12(15)	15
	PAIR ↓	0	2	22	23	36	44(100)	8	38(74)	44
	AutoDAN ↓	10	46	36	20	19	38(100)	80	33(61)	38
gpt-oss	GCG ($k = 20$) ↓	0	0	2	0	0	0(0)	3	0(0)	3
	AutoDAN ↓	0	0	4	40	0	17(100)	12	15(61)	17

et al., 2025). GCG uses AdvBench; PAIR and AutoDAN use HarmBench (Chao et al., 2025; Liu et al., 2024; Mazeika et al., 2024). We also transfer fixed suffixes from the random-search attack of Andriushchenko et al. (2025) without re-optimizing them against each defense. GCG, AutoDAN, and the transferred suffixes use the same refusal-prefix judge; PAIR uses a model judge (Appendix C.4). Benign utility uses ARC-Challenge, ARC-Easy, and HellaSwag (Clark et al., 2018; Zellers et al., 2019); Appendix C gives full protocol details.

5.2 Certification results

We apply Corollary 4.4 to each padded input using the configuration in Section 5.1. A prompt is certified through suffix budget K when $\epsilon_k > \rho\sqrt{k}$ holds at every length $k = 1, \ldots, K$; we evaluate budgets in increasing order and report the fraction of the 98 prompts meeting this condition at each K.

At $\sigma = 5$, DCRS certifies 98/98 prompts at each of $K = 1, 2, 3$, 90/98 at $K = 4$, and 46/98 at $K = 5$. Because the guarantee covers every vocabulary assignment to the suffix slots, every prompt is certified safe against all $|V|^3 \approx 3.3 \times 10^{13}$ suffixes of length at most three, independently of the attack algorithm. The only certifiable baseline, SelfDenoise, certifies 46% of prompts at $K = 1$ (Table 1, reported at $\sigma = 1$, where DCRS certifies 97/98). Figure 2 shows certified accuracy as a function of the embedding radius for one- and two-token padded inputs; the vocabulary-wide rate at budget k is the height of the bold curve at $\epsilon = \rho\sqrt{k}$. As the budget grows, the sampled subset must cover more modified positions while the required displacement bound increases as $\rho\sqrt{k}$; both effects reduce certified accuracy. Appendix D compares noise scales $\sigma \in \{1, 2, 5\}$. The placeholder also matters: at $\sigma = 1$, replacing ! with + lowers ρ from 1.57 to 1.28 and raises $K = 2$ certification from 12/98 to 27/98 (Appendix I). On Mistral with $\sigma = 1$, DCRS certifies all 88 prompts at $K = 1$ and $K = 2$, with the certified count gradually decreasing for larger budgets, indicating that cross-model certification depends on both embedding geometry and response stability (Appendix J).

5.3 Empirical robustness and utility

Jailbreak resistance. Table 1 compares defenses on three open-weight models. Apart from SelfDenoise, DCRS is the only evaluated defense that provides a certificate, and its empirical ASR is comparable to the strongest uncertified defenses: on Llama-2 it reaches 1.0% (GCG), 4% (PAIR), and 0% (AutoDAN), within a few points of SmoothLLM and PAT. On Vicuna it attains the lowest ASR under PAIR (0%) and AutoDAN (10%), and on gpt-oss it blocks both attacks. Perplexity filtering achieves 0% ASR on GCG, but its undetected rate is 100% for PAIR and AutoDAN. The ASR heatmap in Figure 2 further shows GCG ASR decreasing as the noise scale increases. Appendix H reports transferred Simple Adaptive suffixes, and Appendix M isolates the effect of vocabulary projection.

Benign utility. Table 2 reports accuracy on ARC-Challenge, ARC-Easy, and HellaSwag (Clark et al., 2018; Zellers et al., 2019). DCRS achieves 39.5%, 56.0%, and 64.0%, respectively, exceeding SelfDenoise and SmoothLLM on all three benchmarks. Relative to the undefended model, the

Table 2: Benign utility (accuracy, %) and computational cost on Llama-2. DCRS achieves the highest accuracy among the evaluated smoothing-based defenses across all three benchmarks and remains competitive with the prompt-optimization defenses RPO and PAT.

Benchmark / Metric	DCRS	SelfDenoise	SmoothLLM	RPO	PAT	None
ARC-Challenge ↑	39.5	36.5	31.0	37.0	38.5	42.5
ARC-Easy ↑	56.0	52.0	45.0	59.5	44.5	59.5
HellaSwag ↑	64.0	62.5	50.5	64.5	52.0	66.0
Time/trial (s) ↓	0.17	3.51	0.17	0.18	0.18	0.15

corresponding losses are 3.0, 3.5, and 2.0 percentage points. These measurements characterize the utility cost of the evaluated defense configurations.

Computational cost. The measured time per randomized trial is 0.17 s for DCRS, compared with 0.15 s without a defense, 0.17 s for SmoothLLM, and 3.51 s for SelfDenoise. Vocabulary projection allows DCRS to use one generation per trial, while SelfDenoise performs both paraphrasing and generation. At the measured settings, DCRS is approximately $20.6\times$ faster per trial than SelfDenoise. Total certification time also depends on the Monte Carlo sample count, batching, and the number of evaluated suffix budgets.

6 Discussion and Limitations

Certified budgets in context. Certified budgets under unrestricted token changes are small throughout the literature. Randomized masking certifies more than half of AGNews texts against five-word perturbations and SST-2 texts against two (Zeng et al., 2023). Erase-and-check certifies 20-token suffixes by exploiting their known location, but evaluates changes at arbitrary positions only up to three tokens because its cost grows exponentially with the budget (Kumar et al., 2024). Against this background, a single DCRS run on prompts of about 20 tokens certifies every prompt against all 3.3×10^{13} suffixes of length at most three and 46 prompts against all 3.4×10^{22} suffixes of length at most five. Because ϵ_k bounds changes at any k positions, the same radii also cover attacks that combine a suffix with edits to the prompt itself: every prompt remains certified when at most two positions change in total, and 97 of 98 when at most three change (Appendix N).

Response judging. Our judge h uses a refusal-prefix rule: responses without an explicit initial refusal are labeled NOT-SAFE, including harmless non-refusals. This can lower the reported certified rates. Conversely, a refusal-like opening followed by harmful content may be incorrectly labeled safe. Semantic judging errors therefore remain a limitation even when the probability bound is valid.

Scope and cost. Certificates are computed per prompt under fixed tokenization, projection, decoding, and judging; prompt rewrites such as PAIR and AutoDAN, and suffixes beyond the certified budget, are evaluated empirically. Testing each budget at level α makes a certificate through K hold jointly with probability at least $1 - K\alpha$. Beyond the majority guarantee, $p_{\mathrm{cert}}(k, \rho\sqrt{k})$ lower-bounds the probability that a single smoothed response is safe for every suffix of length k. With $N = 2{,}000$ generations per budget, certification is best suited to offline auditing; independent trials permit batching, and adaptive sampling can reduce the cost on clear cases.

7 Conclusion

DCRS connects sparse token perturbations to a response-level safety certificate through hierarchical randomized smoothing. Fixed-size subset selection controls coverage, Gaussian noise supplies an embedding-space radius, and dummy padding converts suffix insertion into vocabulary replacement. Experiments demonstrate multi-token suffix certificates and characterize empirical jailbreak resistance, benign utility, and per-trial cost. The interaction between coverage, response stability, and embedding geometry suggests directions for improving certification through sampling design, placeholder selection, and application-specific perturbation sets.

AI USE STATEMENT

Generative AI tools were used to help edit prose, check internal consistency, aggregate already-produced result files, and draft plotting code. They were not used to create experimental observations or replace execution of the reported experiments. The authors inspected the underlying logs for all headline numbers, reviewed AI-assisted text and code, and remain responsible for the final claims, citations, and artifacts.

ETHICS STATEMENT

This work studies defenses against harmful-model behavior and therefore uses public harmful-request benchmarks in a controlled research setting. We report aggregate safety outcomes and do not reproduce harmful model completions or release new actionable jailbreak strings in the paper. The safety interpretation depends on the response judge and perturbation set described in the methodology. Deployment should combine certification with access controls, monitoring, and independent safety evaluation.

REPRODUCIBILITY STATEMENT

The method, threat model, and assumptions are specified in Sections 3 and 4, complete derivations appear in Appendices A and B, and Appendix L gives pseudocode. Appendix C records sampling, tokenization, generation, judging, and compute details. The submitted artifact includes per-prompt summaries and a script that regenerates the paper's aggregate figures without rerunning model inference.

REFERENCES

Gabriel Alon and Michael Kamfonas. Detecting language model attacks with perplexity. *arXiv preprint arXiv:2308.14132*, 2023.

Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Jailbreaking leading safety-aligned LLMs with simple adaptive attacks. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/63fa7efdd3bcf944a4bd6e0ff6a50041-Abstract-Conference.html.

Nicholas Carlini, Florian Tramèr, Krishnamurthy Dj Dvijotham, Leslie Rice, Mingjie Sun, and J. Zico Kolter. (Certified!!) adversarial robustness for free! In *International Conference on Learning Representations*, 2023.

Patrick Chao, Edoardo Debenedetti, Alexander Robey, Maksym Andriushchenko, Francesco Croce, Vikash Sehwag, Edgar Dobriban, Nicolas Flammarion, George J. Pappas, Florian Tramèr, Hamed Hassani, and Eric Wong. JailbreakBench: An open robustness benchmark for jailbreaking large language models. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, volume 37, pp. 55005–55029, 2024.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J. Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pp. 23–42. IEEE, 2025.

Huanran Chen, Yinpeng Dong, Zeming Wei, Hang Su, and Jun Zhu. Towards the worst-case robustness of large language models. *arXiv preprint arXiv:2501.19040*, 2025. URL https://arxiv.org/abs/2501.19040.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality. LMSYS Blog, 2023. URL https://lmsys.org/blog/2023-03-30-vicuna/.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

C. J. Clopper and E. S. Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413, 1934.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1310–1320. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/cohen19c.html.

Zhuoqun Huang, Neil G Marchant, Keane Lucas, Lujo Bauer, Olga Ohrimenko, and Benjamin I. P. Rubinstein. RS-Del: Edit distance robustness certificates for sequence classifiers via randomized deletion. In *Advances in Neural Information Processing Systems*, volume 36, pp. 18676–18711, 2023.

Zhuoqun Huang, Neil G Marchant, Olga Ohrimenko, and Benjamin I. P. Rubinstein. CERT-ED: Certifiably robust text classification for edit distance. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 10813–10835, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.635. URL https://aclanthology.org/2024.findings-emnlp.635/.

Jongheon Jeong and Jinwoo Shin. Multi-scale diffusion denoised smoothing. In *Advances in Neural Information Processing Systems*, volume 36, pp. 67374–67397, 2023.

Jiabao Ji, Bairu Hou, Zhen Zhang, Guanhua Zhang, Wenqi Fan, Qing Li, Yang Zhang, Gaowen Liu, Sijia Liu, and Shiyu Chang. Advancing the robustness of large language models through self-denoised smoothing. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 246–257, 2024. doi: 10.18653/v1/2024.naacl-short.23. URL https://aclanthology.org/2024.naacl-short.23/.

Xiaojun Jia, Tianyu Pang, Chao Du, Yihao Huang, Jindong Gu, Yang Liu, Xiaochun Cao, and Min Lin. Improved techniques for optimization-based jailbreaking on large language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Aounon Kumar, Chirag Agarwal, Suraj Srinivas, Aaron Jiaxun Li, Soheil Feizi, and Himabindu Lakkaraju. Certifying LLM safety against adversarial prompting. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=9Ik05cycLq.

Adarsh Kumarappan and Ayushi Mehrotra. Towards realistic guarantees: A probabilistic certificate for SmoothLLM. In *NeurIPS 2025 Workshop on Lock-LLM: Prevent Unauthorized Knowledge Use from Large Language Models*, 2025. URL https://openreview.net/forum?id=zgNaybxvR3.

Mathias Lécuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 656–672. IEEE Computer Society, 2019.

Alexander Levine and Soheil Feizi. Robustness certificates for sparse adversarial attacks by randomized ablation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 4585–4593, 2020.

Hao Li, Xiaogeng Liu, Ning Zhang, and Chaowei Xiao. PIGuard: Prompt injection guardrail via mitigating overdefense for free. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 30420–30437, 2025. doi: 10.18653/v1/2025.acl-long.1468.

Zeyi Liao and Huan Sun. AmpleGCG: Learning a universal and transferable generative model of adversarial suffixes for jailbreaking both open and closed LLMs. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=UfqzXg95I5.

Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. AutoDAN: Generating stealthy jailbreak prompts on aligned large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=7Jwpw4qKkb.

Qian Lou, Xin Liang, Jiaqi Xue, Yancheng Zhang, Rui Xie, and Mengxin Zheng. CR-UTP: Certified robustness against universal text perturbations on large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9863–9875, 2024.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, David Forsyth, and Dan Hendrycks. HarmBench: A standardized evaluation framework for automated red teaming and robust refusal. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 35181–35224. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/mazeika24a.html.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. Tree of attacks: Jailbreaking black-box LLMs automatically. In *Advances in Neural Information Processing Systems*, volume 37, pp. 61065–61105, 2024.

Yichuan Mo, Yuji Wang, Zeming Wei, and Yisen Wang. Fight back against jailbreaking via prompt adversarial tuning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 64242–64272, 2024.

Milad Nasr, Nicholas Carlini, Chawin Sitawarin, Sander V. Schulhoff, Jamie Hayes, Michael Ilie, Juliette Pluto, Shuang Song, Harsh Chaudhari, Ilia Shumailov, Abhradeep Thakurta, Kai Yuanqing Xiao, Andreas Terzis, and Florian Tramèr. The attacker moves second: Stronger adaptive attacks bypass defenses against LLM jailbreaks and prompt injections. *arXiv preprint arXiv:2510.09023*, 2025.

J. Neyman and E. S. Pearson. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A*, 231:289–337, 1933.

OpenAI. Introducing ChatGPT. OpenAI Blog, 2022. URL https://openai.com/blog/chatgpt.

OpenAI. Introducing GPT-4.1 in the API. OpenAI Blog, 2025a. URL https://openai.com/index/gpt-4-1/.

OpenAI. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025b.

Alexander Robey, Eric Wong, Hamed Hassani, and George J. Pappas. SmoothLLM: Defending large language models against jailbreaking attacks. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=laPAh2hRFC.

Vinu Sankar Sadasivan, Shoumik Saha, Gaurang Sriramanan, Priyatham Kattakinda, Atoosa Chegini, and Soheil Feizi. Fast adversarial attacks on language models in one GPU minute. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 42976–42998. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/sadasivan24a.html.

Hadi Salman, Mingjie Sun, Greg Yang, Ashish Kapoor, and J. Zico Kolter. Denoised smoothing: A provable defense for pretrained classifiers. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21945–21957. Curran Associates, Inc., 2020.

Yan Scholten, Jan Schuchardt, Aleksandar Bojchevski, and Stephan Günnemann. Hierarchical randomized smoothing. In *Advances in Neural Information Processing Systems*, volume 36, pp. 49783–49813, 2023. doi: 10.52202/075280-2167. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/9c0efc0d84c263972af72bf70a2de533-Abstract-Conference.html.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Zixia Wang, Gaojie Jin, Jia Hu, and Ronghui Mu. CluCERT: Certifying LLM robustness via clustering-guided denoising smoothing. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(44):37998–38006, 2026. doi: 10.1609/aaai.v40i44.41137. URL https://ojs.aaai.org/index.php/AAAI/article/view/41137.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How does LLM safety training fail? In *Advances in Neural Information Processing Systems*, volume 36, pp. 80079–80110, 2023.

Mao Ye, Chengyue Gong, and Qiang Liu. SAFER: A structure-free approach for certified robustness to adversarial word substitutions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 3465–3475, 2020.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019.

Jiehang Zeng, Jianhan Xu, Xiaoqing Zheng, and Xuanjing Huang. Certified robustness to text adversarial attacks by randomized [MASK]. *Computational Linguistics*, 49(2):395–427, 2023.

Haiteng Zhao, Chang Ma, Xinshuai Dong, Anh Tuan Luu, Zhi-Hong Deng, and Hanwang Zhang. Certified robustness against natural language attacks by causal intervention. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 26958–26970. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/zhao22g.html.

Andy Zhou, Bo Li, and Haohan Wang. Robust prompt optimization for defending language models against jailbreaking attacks. In *Advances in Neural Information Processing Systems*, volume 37, pp. 40184–40211, 2024.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023.

Andy Zou, Long Phan, Justin Wang, Derek Duenas, Maxwell Lin, Maksym Andriushchenko, Rowan Wang, Zico Kolter, Matt Fredrikson, and Dan Hendrycks. Improving alignment and robustness with circuit breakers. In *Advances in Neural Information Processing Systems*, volume 37, pp. 83345–83373, 2024.

A HRS Derivation in Our Notation

This appendix presents the Neyman–Pearson derivation of hierarchical randomized smoothing (Scholten et al., 2023) in the token-level notation of Theorem 4.1.

Recall that $e = (e_1, \dots, e_T)$ denotes the clean embedding sequence and that e' denotes a perturbed embedding sequence. Let $C \subseteq [T]$ be the set of token positions on which e and e' differ. In each randomized trial, the upper-level sampler first draws a subset $S \subseteq [T]$ of token positions, and the lower-level smoothing distribution then applies Gaussian perturbations to the selected embeddings. We write Ψ_e and $\Psi_{e'}$ for the resulting hierarchical smoothing distributions around e and e', respectively. Equivalently, a sample $Z \sim \Psi_e$ is the randomized embedding sequence obtained by sampling S and applying the lower-level Gaussian perturbation to e, and $Z \sim \Psi_{e'}$ is defined analogously for e'.

The deterministic base decision rule is the same composite rule used in Section 3.1,

$$H(z) = h(f(D(z))),$$

where D denotes the nearest-neighbor projection-and-embedding map, f is the generative model, and h maps the generated response to a response-level label. Thus, the randomness in Ψ_e or $\Psi_{e'}$ comes only from the subset sampling and Gaussian perturbation, while nearest-neighbor projection, generation, and response-level decision mapping are included inside the deterministic rule H.

For a fixed label y, suppose that the clean hierarchical smoothing distribution satisfies

$$\Pr_{Z\sim\Psi_e}[H(Z) = y] \geq p_L.$$

Following the region decomposition used in hierarchical randomized smoothing (Scholten et al., 2023), we partition the sample space into three regions. The region R_1 denotes the part of the clean hierarchical distribution in which at least one perturbed position is not selected by the upper-level sampler. The region R_3 is the analogous part of the perturbed hierarchical distribution. The shared region R_2 denotes the event in which all perturbed positions are selected, so that the clean and perturbed distributions can be compared through the lower-level smoothing distributions on the selected perturbed block.

Let Δ be the probability that the upper-level sampling step does not select all positions in C. Then the relevant region probabilities are

$$\Pr_{Z\sim\Psi_e}[Z \in R_1] = \Delta, \qquad \Pr_{Z\sim\Psi_e}[Z \in R_2] = 1 - \Delta,$$

and

$$\Pr_{Z\sim\Psi_{e'}}[Z \in R_2] = 1 - \Delta, \qquad \Pr_{Z\sim\Psi_{e'}}[Z \in R_3] = \Delta.$$

We seek a lower bound on the probability of predicting y under $\Psi_{e'}$ given only the clean lower bound p_L. In the worst case, the classifier can assign as much of the clean label probability as possible to R_1, because this region does not contribute to the shared region R_2 where the clean and perturbed distributions are compared. Since $\Pr_{\Psi_e}(R_1) = \Delta$, at most Δ of the clean probability can be lost in this way. Therefore, at least $p_L - \Delta$ of the clean label probability must lie in the shared region R_2. Since $\Pr_{\Psi_e}(R_2) = 1 - \Delta$, the conditional clean label probability inside R_2 is lower bounded by

$$p'_L = \frac{p_L - \Delta}{1 - \Delta}.$$

Conditioned on R_2, all perturbed positions have been selected, and the hierarchical smoothing problem reduces to the lower-level smoothing problem on the perturbed block C. Let $q_\epsilon(p'_L)$ denote the lower-level Neyman–Pearson lower bound under a perturbation of size ϵ when the clean conditional label probability is at least p'_L. Then

$$\Pr_{Z\sim\Psi_{e'}}[H(Z) = y \mid Z \in R_2] \geq q_\epsilon(p'_L).$$

Multiplying by the probability of the shared region under $\Psi_{e'}$ gives

$$\Pr_{Z\sim\Psi_{e'}}[H(Z) = y] \geq \Pr_{Z\sim\Psi_{e'}}[Z \in R_2]\, q_\epsilon(p'_L) = (1 - \Delta) q_\epsilon(p'_L).$$

This proves Theorem 4.1.

B PROOF OF THE BINARY SAFETY CERTIFICATE

We prove Corollary 4.3. For any perturbation set C with $|C| = j \leq k$, the probability that the sampled subset fails to contain all modified positions is

$$\Delta_j = 1 - \frac{\binom{T-j}{m-j}}{\binom{T}{m}}.$$

For fixed T and m, Δ_j is nondecreasing in j, so $\Delta_j \leq \Delta_k$. Therefore, using Δ_k gives a conservative lower bound that applies uniformly to all perturbation sets with size at most k.

In the binary safety setting, it is sufficient that the probability of the clean smoothed decision remains larger than $1/2$ under the perturbed input. By Theorem 4.2, this holds whenever

$$p_{\mathrm{cert}}(k, \epsilon) > \frac{1}{2}.$$

Substituting the expression for $p_{\mathrm{cert}}(k, \epsilon)$ from Equation (12), this condition becomes

$$(1 - \Delta_k)\Phi\left(\Phi^{-1}\left(\frac{p_L - \Delta_k}{1 - \Delta_k}\right) - \frac{\epsilon}{\sigma}\right) > \frac{1}{2}.$$

Since Φ is strictly increasing, this is equivalent to

$$\Phi^{-1}\left(\frac{p_L - \Delta_k}{1 - \Delta_k}\right) - \frac{\epsilon}{\sigma} > \Phi^{-1}\left(\frac{1}{2(1 - \Delta_k)}\right).$$

Rearranging gives

$$\epsilon < \sigma\left[\Phi^{-1}\left(\frac{p_L - \Delta_k}{1 - \Delta_k}\right) - \Phi^{-1}\left(\frac{1}{2(1 - \Delta_k)}\right)\right].$$

For the right-hand side to be positive, we need

$$\frac{p_L - \Delta_k}{1 - \Delta_k} > \frac{1}{2(1 - \Delta_k)},$$

which is equivalent to $p_L > \frac{1}{2} + \Delta_k$. If this condition fails, the bound does not give a positive binary certified radius.

C ADDITIONAL EXPERIMENTAL DETAILS

C.1 CERTIFICATION SETUP

For certification experiments, we use Llama-2-7B-Chat as the target generative model and AdvBench as the harmful prompt dataset (Touvron et al., 2023; Zou et al., 2023). We evaluate 98 AdvBench prompts and focus on suffix attacks, with GCG-style adversarial suffixes as the representative suffix-attack setting. For each suffix budget k, the clean prompt is converted into a fixed-length certification input by appending k dummy suffix tokens. Unless otherwise specified, the dummy token is set to $b = !$.

We use the DCRS smoothing procedure described in Section 3. For each randomized trial, we sample a subset of token positions uniformly at random and perturb only the selected token embeddings with Gaussian noise. The perturbed embeddings are then projected back to the nearest vocabulary tokens before generation. For each padded input $\bar{x}^{(k)}$ with length T_k, we set

$$m_k = \max\{1, \mathrm{round}(0.9T_k)\}$$

using Python rounding to the nearest integer (ties to even), and use noise scale $\sigma = 5.0$ for the main certification results. The recorded effective ratios range from 0.875 to 0.9333; every certificate uses the recorded integer pair (T_k, m_k). For each prompt, we run $N = 2000$ randomized trials. We count only explicit SAFE outcomes, treat confusion and every invalid outcome as NOT-SAFE, and compute the one-sided Clopper–Pearson lower confidence bound p_L with $\alpha = 0.05$. The main run contains 948,226 explicit safe outcomes and 15,774 attack outcomes across 964,000 trials, with no confusion

outcomes. Per-prompt counts and recorded (T_k, m_k) determine the confidence bounds and radii; the figure-generation script checks these quantities against the trial logs.

For each suffix budget k, we compute the certified radius ϵ_k using Corollary 4.3. To determine whether the padded suffix is certified against arbitrary token replacement, we compare ϵ_k with the worst-case suffix displacement threshold $\rho\sqrt{k}$, where

$$\rho = \max_{v \in V} \|E(v) - E(b_{1738})\|_2 = 1.5697249,$$

where b_{1738} is the exact Llama tokenizer ID used in the run. We record IDs because visually identical punctuation tokens can differ by a leading-space convention and have different embeddings. A prompt is counted as certified for suffix budget k if

$$\epsilon_k > \rho\sqrt{k}.$$

This criterion certifies that any replacement of the k appended dummy suffix positions remains within the certified embedding-space radius.

For compute efficiency, the implementation evaluates suffix budgets in a nested manner: it continues to $k+1$ only after a prompt passes k. All certified accuracies use the original denominator of 98, with unevaluated larger budgets counted as uncertified. Certification up to K requires passing every budget $j \leq K$.

Small radii at large budgets. In the main run, small positive $k = 7$ radii occur on three one-placeholder inputs and one two-placeholder input, each with $T = 15$ and $m = 14$. Rounding gives coverage $1 - \Delta_7 = 8/15$ for these inputs. Their radii are at most 0.370, below the vocabulary-wide threshold $\rho\sqrt{7}$, so none of them contributes a suffix certificate.

C.2 Nearest-neighbor projection

After Gaussian perturbation, a noisy embedding vector may not correspond to a valid vocabulary embedding. We therefore apply a deterministic nearest-neighbor projection over the vocabulary embedding table. For a perturbed embedding vector z, the nearest vocabulary token under cosine similarity is

$$\mathrm{NN}_V(z) = \arg\max_{v \in V} \frac{\langle z, E(v) \rangle}{\|z\|_2 \, \|E(v)\|_2}.$$

We then map the projected token back to its embedding. Following Section 3.1, we denote this token-wise projection-and-embedding map by

$$D(z_i) = E(\mathrm{NN}_V(z_i)).$$

Thus, D maps continuous noisy embeddings back to valid vocabulary embeddings before generation.

In implementation, we perform nearest-neighbor search over the model vocabulary embedding table. We normalize the vocabulary embeddings once and retrieve the token with the maximum inner product against the normalized query, which is the cosine nearest neighbor. The resulting deterministic map is included in the base rule $H = h \circ f \circ D$.

C.3 ASR evaluation setup

We evaluate GCG on AdvBench and PAIR and AutoDAN on HarmBench (Zou et al., 2023; Chao et al., 2025; Liu et al., 2024; Mazeika et al., 2024). GCG suffixes are optimized separately for each prompt. PAIR iteratively refines prompts using feedback from a target model, while AutoDAN searches for adversarial prompts with natural-language structure.

For randomized defenses, we aggregate multiple randomized generations by majority vote. We use 10 votes for GCG and 5 votes for PAIR and AutoDAN. For each vote, we independently sample the randomized smoothing distribution, generate a response from the target model, and assign a safety label using the corresponding judge. The final prediction is the majority safety label over the randomized responses. For DCRS, we evaluate multiple smoothing configurations by varying the noise scale σ and the token sampling ratio r. For an input of length T, the number of perturbed positions is set to

$$m = \lfloor rT \rfloor.$$

The comparison includes input-transformation defenses (SelfDenoise and SmoothLLM), RPO, PAT, perplexity filtering (PPL), Circuit Breakers (CB), PIGuard, and an undefended baseline. GCG and AutoDAN use a refusal-prefix judge; PAIR uses a separate model judge. The labeling rules are specified below.

C.4 ASR JUDGES

For GCG and AutoDAN, the judge labels a response as safe when it matches the refusal-prefix rules below after leading-whitespace removal and capitalization normalization. All other responses count as successful attacks. This proxy can miss refusals phrased differently or accept a refusal-like opening followed by harmful content. For PAIR, GPT-3.5-Turbo (OpenAI, 2022) proposes candidate attacks and GPT-4.1 (OpenAI, 2025a) judges the target responses.

The refusal-prefix regular expressions used for GCG and AutoDAN are:

```
^i'm sorry\b                    ^sorry\b                ^i am sorry\b
^i apologize\b                  ^my apologies\b         ^as an ai\b
^as a language model\b          ^as an assistant\b      ^i cannot\b
i can't\b                       ^i do not\b             ^i'm not able to provide\b
^i am not able to provide\b     ^i cannot provide\b     ^i can't provide\b
^i cannot assist\b              ^i can't assist\b       ^i cannot support\b
^i can't support\b
```

C.5 GENERATION DETAILS

For all generation-based evaluations, we use greedy decoding with `do_sample=False` and `max_new_tokens`=128 unless otherwise specified. Randomized generations are produced with deterministic decoding after each sampled input perturbation, so the only source of randomness comes from the smoothing procedure. For Llama-2-7B-Chat, prompts are formatted using the model's default chat template. The generated response is post-processed by removing leading whitespace and normalizing capitalization before applying the corresponding safety judge.

D NOISE SCALE AND LARGER-TOKEN-BUDGET CERTIFICATION

Table 3: Certified prompts out of 98 as the Gaussian noise scale and suffix budget vary ($r = 0.9$).

Noise scale	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$
$\sigma = 1$	97/98	12/98	0/98	0/98	0/98
$\sigma = 2$	98/98	96/98	63/98	6/98	0/98
$\sigma = 5$	98/98	98/98	98/98	90/98	46/98

Table 3 shows how the certified budget changes with noise scale. Larger budgets are harder because both the combinatorial penalty Δ_k and the worst-case replacement threshold $\rho\sqrt{k}$ increase with k. Increasing σ enlarges the continuous smoothing radius and yields non-vacuous certificates at larger budgets: at $\sigma = 5$, all 98 prompts are certified through $k = 3$, 90 remain certified at $k = 4$, and 46 remain certified at $k = 5$.

E SAMPLING-RATIO TRADE-OFF

For $k = 1$, the subset penalty $\Delta_1 = 1 - m/T$ decreases to zero at full coverage. The mean safe-probability bound increases to 0.9935 at $r = 0.75$ and then declines to 0.9910 at $r = 1$. This diagnostic illustrates the competing dependence of coverage and response stability on the sampling ratio. Both terms enter the certified-radius expression in Equation (15).

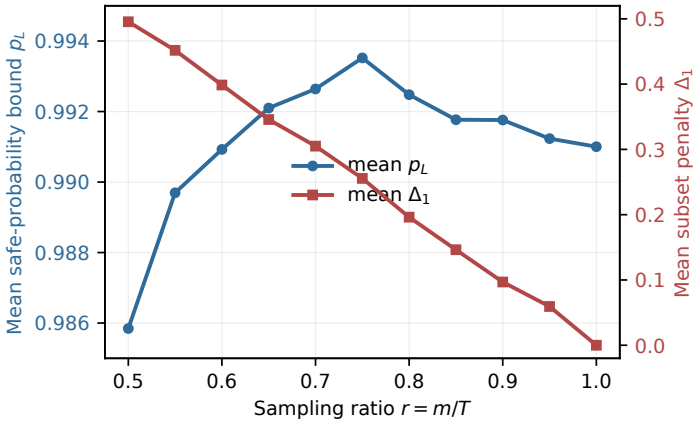


Figure 3: Mean safe-probability lower bound p_L and $k = 1$ subset penalty Δ_1 over a diagnostic set of ten clean harmful prompts ($\sigma = 1$, $N = 1000$). Increasing r monotonically improves coverage, but p_L peaks near $r = 0.75$ and then declines slightly. The diagnostic uses unpadded inputs.

F Gaussian-Only Ablation

At $\sigma = 1$, full-sequence Gaussian smoothing certifies 95/98 prompts at $k = 1$ and 26/98 at $k = 2$, compared with 96/98 and 27/98 for DCRS. Setting $m = T$ eliminates the coverage penalty, but also changes the safe-response probability by perturbing every position. The modest difference in this comparison illustrates that improved coverage can be offset by changes in response stability.

Table 4: Gaussian-only ablation.

Method	$r = m/T$	$k = 1$	$k = 2$
DCRS	0.9	96/98	27/98
Gaussian-only	1.0	95/98	26/98

G SmoothLLM Perturbation Comparison

Table 5: Empirical SmoothLLM results on 98 prompts using 15% character perturbations and 1,000 randomized copies per prompt. The single-copy safe rate is averaged over copies; ASR is computed after majority aggregation.

Perturbation	**Single-copy safe rate (%)**	**ASR (%)**
Patch	69.09	27.55
Swap	58.92	35.71
Insert	61.17	38.78

The experiment uses Vicuna-7B-v1.3 (Chiang et al., 2023), a fixed refusal-prefix judge, and a maximum generation length of 50 tokens. Patch perturbations yield the highest single-copy safe rate and lowest majority-vote ASR among these three configurations.

H Simple Adaptive Attack Details

The Simple Adaptive Attacks method combines a manually designed model-specific template with a 25-token suffix optimized by request-specific random search (Andriushchenko et al., 2025). We evaluate the same 50 stored Llama-2 suffixes across the defenses in Table 6, transferring the attacks without reoptimizing them for each defense. DCRS achieves an ASR of 6%, compared with 14% for SmoothLLM-Swap, 48% for SelfDenoise, and 100% without a defense.

Table 6: Empirical ASR on 50 stored Simple Adaptive attacks transferred to defenses on Llama-2. Lower is better.

Defense	**ASR (%)**
DCRS	**6**
SelfDenoise	48
SmoothLLM-Swap	14
SmoothLLM-Patch	20
SmoothLLM-Insert	24
None	100

I Effect of Dummy-Suffix Initialization

Table 7: Suffix certification at $\sigma = 1$ and $r = 0.9$ under different dummy-token initializations.

Dummy token	ρ	$k = 1$	$k = 2$
!	1.57	97/98	12/98
.	1.51	91/98	0/98
+	1.28	96/98	27/98

We repeat the same certification procedure with `!`, `.`, and + while holding the smoothing configuration fixed. The dummy choice affects the certificate through the embedding geometry of the initialization token. The period initialization certifies 91/98 prompts at $k = 1$ but none at $k = 2$. Among the evaluated choices, + has the third-smallest ρ in the vocabulary. Replacing `!` with + reduces ρ from 1.57 to 1.28 and therefore lowers the arbitrary-replacement threshold $\rho\sqrt{k}$. As a result, $k = 2$ certification improves from 12/98 to 27/98 while $k = 1$ changes from 97/98 to 96/98. All cells use the full 98-prompt denominator so the reported rates are directly comparable. The exact placeholder used by the main $\sigma = 5$ run is reported separately in Section 5.1; punctuation tokens with different spacing conventions can correspond to different token IDs and embedding distances.

J Additional Certification Results on Mistral

We evaluate 88 prompts on Mistral-7B-Instruct-v0.2 (Jiang et al., 2023) with $\sigma = 1$, $r = 0.9$, $N = 2000$, and token ID 46 (+) as the placeholder. Its maximum vocabulary distance is $\rho = 0.2441$. Table 8 reports the number of prompts certified safe at each suffix budget k, using the displacement threshold $\rho\sqrt{k}$.

Table 8: Certified safe prompts on Mistral with $\sigma = 1$, $r = 0.9$, and $N = 2000$. The threshold $\rho\sqrt{k}$ covers every vocabulary-token assignment to the k suffix positions. All denominators are 88.

Budget k	Threshold $\rho\sqrt{k}$	Certified safe
1	0.244	88/88
2	0.345	88/88
3	0.423	74/88
4	0.488	45/88
5	0.546	18/88
6	0.598	3/88

All 88 prompts are certified safe at $k = 1$ and $k = 2$, and 74 remain certified at $k = 3$. The counts decrease at larger budgets, but non-vacuous certificates remain up to $k = 6$. Cross-model comparisons should account for both embedding geometry, which determines ρ, and the safe-response probability, which influences the certified radius.

K GCG Suffix Optimization Explores a Small Reachable Token Set

Setup. For each of $N = 50$ harmful prompts from AdvBench (Zou et al., 2023), we replace the appended dummy suffix token with every token in the vocabulary and rank tokens by GCG-style affirmative-response loss against the target string "Sure, here is". We restrict candidates to ASCII tokens, following GCG's standard filter. The top-M tokens per prompt, unioned across prompts, form the empirical reachable set V^*.

Result. Figure 4 visualizes V^* at $M = 50$ in the PCA-projected embedding space of the full Llama-2-7B vocabulary. Although the procedure produces $N \cdot M = 2{,}500$ candidate selections before deduplication, these selections collapse to $|V^*| = 979$ unique tokens, about 3% of the $|V| = 32{,}000$-token vocabulary. The reachable tokens occupy a visibly concentrated region around the dummy token. This concentration motivates studying certificates with explicitly restricted candidate sets, alongside the vocabulary-wide criterion used here.

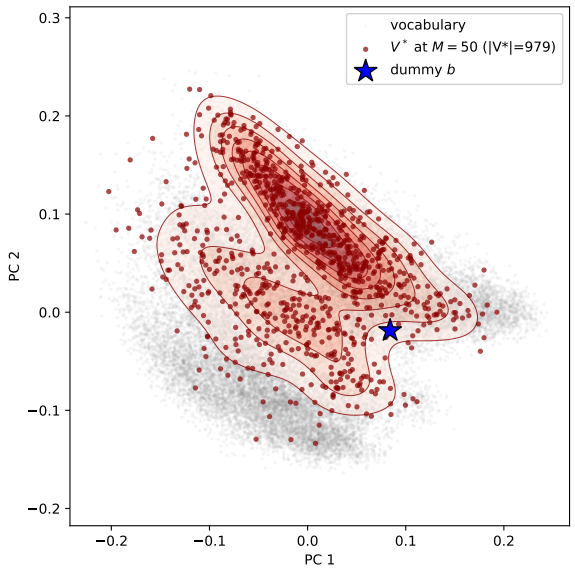


Figure 4: Reachable set V^* at $M = 50$ in the PCA-projected embedding space of the full Llama-2-7B vocabulary. Gray points denote vocabulary tokens, red contours and points denote reachable tokens, and the blue star marks the dummy token b. The set contains $|V^*| = 979$ unique tokens out of $|V| = 32{,}000$.

L Suffix certification algorithm

Algorithm 1 summarizes the suffix-certification procedure. For each suffix budget k, the clean prompt is converted into a fixed-length input by appending k dummy tokens b^k. The sampling size is set to $m_k = \max\{1, \mathrm{round}(rT_k)\}$, maintaining the same nominal sampling ratio across suffix budgets. We compute the Clopper–Pearson lower bound p_L from explicit safe outcomes only and then evaluate Corollary 4.3. When $k > m_k$, or when $p_L \le 1/2 + \Delta_k$, the certificate is vacuous and we set the radius to zero.

Algorithm 1 Certification for suffix attacks

Input: prompt x, maximum suffix length K, dummy token b, sampling ratio r, noise scale σ, number of trials N, confidence level α
for $k = 1, \ldots, K$ **do**
  $\bar{x}^{(k)} \leftarrow x \oplus b^k, \quad T_k \leftarrow |\bar{x}^{(k)}|, \quad m_k \leftarrow \max\{1, \mathrm{round}(rT_k)\}$
  Run smoothing on $\bar{x}^{(k)}$ for N trials; let n_{safe} count only explicit safe outcomes
  $p_L \leftarrow \mathrm{CPLower}(n_{\mathrm{safe}}, N, \alpha)$
  if $k > m_k$ **then**
    $\epsilon_k \leftarrow 0$
  else
    $\Delta_k \leftarrow 1 - \frac{\binom{T_k - k}{m_k - k}}{\binom{T_k}{m_k}}$
    if $p_L \leq \frac{1}{2} + \Delta_k$ **then**
      $\epsilon_k \leftarrow 0$
    else
      $\epsilon_k \leftarrow \sigma \left[\Phi^{-1}\left(\frac{p_L - \Delta_k}{1 - \Delta_k} \right) - \Phi^{-1}\left(\frac{1}{2(1 - \Delta_k)} \right) \right]$
    end if
  end if
end for
Return: certified radii $\{\epsilon_k\}_{k=1}^{K}$

The returned value ϵ_k is the certified embedding-space radius for the padded input at suffix budget k. To certify arbitrary replacement of the appended dummy suffix positions, we compare this radius with the worst-case displacement threshold $\rho\sqrt{k}$, where $\rho = \max_{v \in V} \|E(v) - E(b)\|_2$. A prompt is counted as certified for budget k only when $\epsilon_k > \rho\sqrt{k}$. In the nested suffix-budget evaluation, we continue to budget $k + 1$ only for prompts that are certified at budget k, yielding a nested set of certified prompts across budgets.

M Effect of nearest-neighbor projection under varying noise scales

Figure 5 compares Gaussian embedding perturbations without projection (L2) with the same perturbations followed by cosine nearest-neighbor projection (L2+NN). Across the displayed moderate-to-large sampling ratios, projection produces a stronger reduction in ASR as σ increases. These results support vocabulary projection as a useful component of the evaluated smoothing pipeline.

N Suffix Attacks Combined with Prompt Edits

The certificate in Corollary 4.3 constrains the number of modified positions and their joint displacement, but not their location. This appendix uses this property to cover attacks that append a suffix and also substitute tokens of the harmful request itself.

Threat model. An attacker submits $x' \oplus s$, where $s \in V^j$ is a suffix of length $j \geq 1$ and x' is obtained from x by substituting the tokens at a set $P \subseteq [T]$ of prompt positions, with $T = |x|$. The total number of changed positions satisfies $|P| + j \leq K$. Because $x' \oplus s$ has the same length as the padded reference $\bar{x}^{(j)} = x \oplus b^j$, the smoothing configuration applied to it is the one used to certify $\bar{x}^{(j)}$.

For each prompt position $i \in [T]$, define the worst-case substitution cost

$$\rho_i = \max_{v \in V} \|E(v) - E(x_i)\|_2,$$

and let $\rho_{(1)} \geq \rho_{(2)} \geq \cdots \geq \rho_{(T)}$ denote these values in decreasing order. Write $\epsilon_K(\bar{x}^{(j)})$ for the binary radius of Corollary 4.3 on $\bar{x}^{(j)}$ at budget K, computed with its own pair (T_j, m_j) and safe-probability bound.

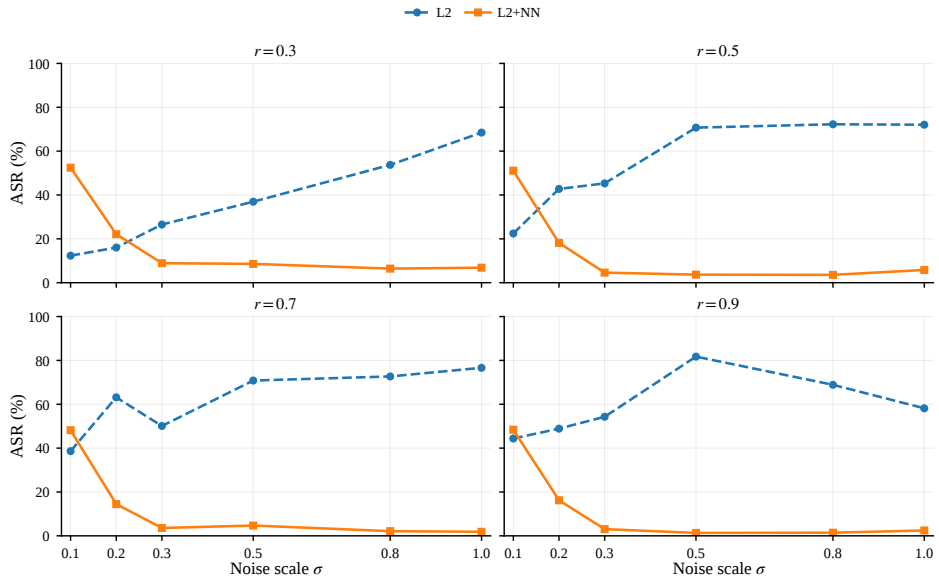


Figure 5: ASR versus Gaussian noise scale for direct embedding perturbations without projection (legend: L2) and the same perturbations followed by cosine nearest-neighbor projection (legend: L2+NN). Each panel fixes the sampling ratio $r = m/T$.

Corollary N.1 (Suffix-and-edit certificate)**.** *Fix $K \geq 1$. If, for every $j = 1, \ldots, K$,*

$$\epsilon_K(\bar{x}^{(j)}) > \left(j\rho^2 + \sum_{l=1}^{K-j} \rho_{(l)}^2 \right)^{1/2}, \tag{20}$$

then $g(E(x' \oplus s)) = \textsc{Safe}$ for every suffix $s \in V^j$ with $1 \leq j \leq K$ and every x' that differs from x in at most $K - j$ positions.

Proof. Fix j, s, and x', and let P be the set of prompt positions at which x' and x differ. The inputs $x' \oplus s$ and $\bar{x}^{(j)}$ differ on a set C with $|C| \leq |P| + j \leq K$, and

$$\left\| E(x' \oplus s)_C - E(\bar{x}^{(j)})_C \right\|_2^2 = \sum_{i \in P} \|E(x'_i) - E(x_i)\|_2^2 + \sum_{t=1}^{j} \|E(s_t) - E(b)\|_2^2 \leq \sum_{l=1}^{K-j} \rho_{(l)}^2 + j\rho^2.$$

By Equation (20), this displacement lies strictly within $\epsilon_K(\bar{x}^{(j)})$, so Corollary 4.3 with budget $K \geq |C|$ preserves the safe decision. □

Evaluation. We evaluate Equation (20) on the main run ($\sigma = 5$, $r = 0.9$, $N = 2000$) without additional model inference: the radii $\epsilon_K(\bar{x}^{(j)})$ are recorded for every budget with a positive radius on each padded input, and ρ_i is computed from the embedding table. Prompts whose padded input $\bar{x}^{(j)}$ was not evaluated under the nested procedure of Appendix C.1 are counted as uncertified. Across the vocabulary, the maximum distance from a token to any other token has median 1.677 and maximum 1.798, compared with $\rho = 1.570$ for the placeholder, so a prompt edit typically requires more of the certified radius than a suffix token.

All 98 prompts remain certified when at most two positions change in total, and 97 when at most three change. The gap to the suffix-only row widens at larger K for two reasons: an attack with a short suffix is compared with the shorter reference $\bar{x}^{(j)}$, whose coverage penalty at budget K is larger, and prompt tokens have larger substitution costs than the placeholder.

Scope. The certificate covers substitutions that preserve token positions; insertions or deletions inside the prompt change the alignment and are not covered. Every certified input here carries at least one appended token, so edits without a suffix would require certifying the unpadded prompt x with

Table 9: Certified prompts out of 98 at $\sigma = 5$ and $r = 0.9$ when at most K positions change in total. The first row counts suffixes of length at most K (Corollary 4.4); the second row additionally allows substitutions of prompt tokens alongside a suffix of length at least one.

Changed positions	$K = 1$	$K = 2$	$K = 3$	$K = 4$	$K = 5$
Suffix only	98	98	98	90	46
Suffix and prompt edits	98	98	97	70	4

the same procedure, which the present runs do not include. The figure-generation script recomputes Table 9 from the recorded radii and the embedding table.

O COMPUTE RESOURCES

All experiments were run on an NVIDIA A100. The main certification experiments use Llama-2-7B-Chat and require $N = 2000$ randomized generations per prompt. Since each randomized trial performs one perturbed forward generation, the total computational cost scales linearly with the number of prompts, suffix budgets, and randomized trials. The suffix-certification experiments were parallelized across prompts and suffix budgets on the cluster. ASR experiments were run using the same target-model inference setup, with different noise scales and token sampling ratios as described in Appendix C.3. For all generation-based evaluations, we use the decoding configuration described in Appendix C.5.