IMPROVING CERTIFIED FEW-SHOT ADAPTATION WITHOUT CHANGING CLEAN PREDICTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Few-shot predictors can be misled by correctly labeled copies of their support examples. Copying changes class counts and can reverse predictions even when class-conditional feature distributions stay fixed. We separate count-dependent coefficients from support statistics and hold these coefficients at their initial values. This normalization preserves every clean prediction for trimmed-distance and exponential-cache scores, with certificates for their complete modified scores. For the cache, we prove that normalization cannot reduce robustness to a single cross-class stored-source copy when all sources are mutable. The remaining copy sensitivity depends on within-class kernel deviations. Across eight text and image encoders, normalization reduces copy harm without changing any clean prediction. On two image tasks, normalized Tip-Adapter retains 87.5% clean accuracy, and 85.4% of predictions remain correct under every single cross-class stored-source copy, compared with 80.4% for zero-shot classification. Under the broader threat of four arbitrary bounded-feature replacements, certified accuracy rises from zero to 28.3%. Fixing initial coefficients can thus preserve adaptation gains while improving resistance to support-set poisoning.

1 INTRODUCTION

Few-shot adaptation specializes pretrained representations through a small labeled support set (Snell et al., 2017; Radford et al., 2021; Zhang et al., 2022a). This set affects predictions even when model weights remain fixed. We study how to improve its poisoning resistance while preserving the adapted predictor's clean decisions. Figure 1 isolates the source-count effect that motivates our approach.

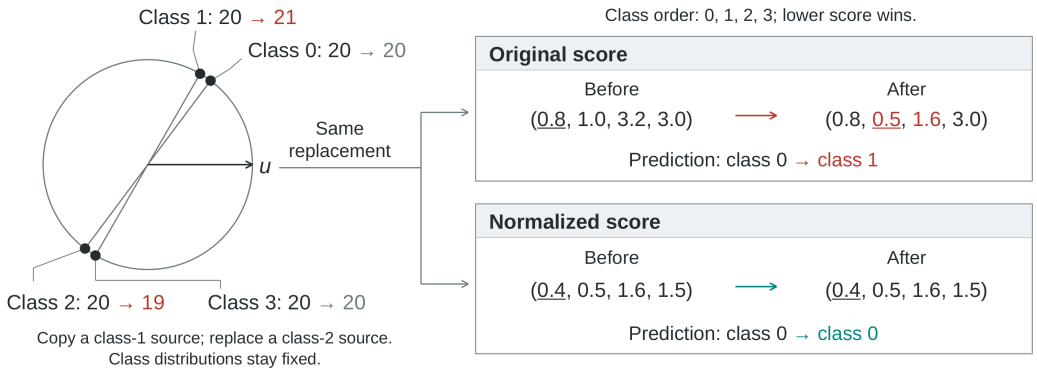


Figure 1: A prediction flip caused by class counts alone. Each class initially contains 20 copies of a fixed feature. Replacing a class-2 record with a correctly labeled class-1 copy changes their counts to 19 and 21 without changing the class-conditional feature distributions. The original prediction changes from class 0 to class 1; the normalized prediction remains class 0. Section 3.2 derives this feature-space construction.

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.

Correct labels do not by themselves make a support set safe. Data poisoning can act through correctly labeled examples, as demonstrated by attacks on learned classifiers (Shafahi et al., 2018; Geiping et al., 2021). For a frozen encoder, replacing one support record has an additional consequence: the total support size can remain fixed while individual class sizes change. This matters when the final score depends on those sizes. A security analysis must therefore specify both which records may change and how the score responds to the resulting counts.

Support aggregation couples feature content to class multiplicity. FCert uses trimmed distance means; Language-empowered Few-shot Certification (LeFCert) combines a retained sum with a retained mean (Wang et al., 2024; Lai et al., 2025). Their guarantees preserve class counts. Extending the latter score to changing counts makes its retained-sum coefficient vary. Tip-Adapter exposes a second instance: its classwise similarity sum scales a cache term relative to fixed text logits (Zhang et al., 2022a). In both cases, changing counts can change the influence of unchanged class statistics.

This coefficient dependence can reverse a prediction without changing class-conditional content. In Figure 1, each class contains identical features. Both rules are evaluated before and after the same replacement. Copying one correctly labeled record between classes leaves their normalized empirical distributions unchanged, yet the original prediction flips while the normalized prediction stays fixed. Sums and means are known to retain different multiset information (Xu et al., 2019). Here that distinction identifies a source of poisoning sensitivity and a direct way to remove it.

Fixing the initial component coefficients preserves the adapted predictor. We express each score as count-dependent weights applied to support statistics, then hold the weights at their initial values. The general correction preserves initial scores for arbitrary positive class counts; the distance specialization additionally uses a common positive rescaling. The statistics can still change after replacement, so we certify the complete score using source-level bounds and a shared replacement budget.

For the cache, we derive an exact condition for robustness to one stored-source copy. The original worst margin loss depends on kernel maxima, whereas the normalized loss depends on deviations from class means in both transfer directions. Constant class kernels give invariance to this copy family; within-class variation determines the remaining sensitivity.

The experiments test this principle across interfaces and against an adaptation-free reference. A fresh panel covers five text and three image encoders, with clean predictions preserved and copy harm reduced for all eight. A further cache study uses two encoders and 40 new support sets. A post hoc analysis finds that normalized adaptation retains more correct predictions than zero-shot classification under every permitted single copy. The broader source-replacement certificates and shared-cache attacks provide separate tests of its poisoning resistance.

2 Related Work

Few-shot methods combine pretrained features with small task-specific support sets. Prototypical networks classify by distances to class representatives (Snell et al., 2017). Contrastive Language-Image Pre-training (CLIP) and Sigmoid loss for Language-Image Pre-training (SigLIP) provide comparable image and text features (Radford et al., 2021; Zhai et al., 2023). CLIP-Adapter learns a feature adapter, while training-free Tip-Adapter builds an exponential cache from labeled support features (Gao et al., 2024; Zhang et al., 2022a). We use the latter's ten-augmentation cache construction with coefficients fixed for our study, and correct class mass under source replacement without changing clean decisions.

Existing few-shot certificates already account for several poisoning budgets. FCert considers both per-class and total-across-class budgets (Wang et al., 2024). LeFCert certifies individual queries; LeFCert-C certifies multiple queries against one shared poisoned support set (Lai et al., 2025). Both method families retain the original class counts. We address changing counts in the maximal-trim two-term score, with a shared replacement budget for each queried prediction. The new object is the count-dependent score and its prediction-preserving normalization; order-statistic bounds and budget accounting supply established proof tools.

Poisoning defenses depend on the allowed intervention and learning procedure. Prior work studies training-data contamination (Biggio et al., 2012; Steinhardt et al., 2017), correct-label attacks

(Shafahi et al., 2018; Geiping et al., 2021), randomized smoothing (Rosenfeld et al., 2020), partition aggregation (Levine & Feizi, 2021), and bagging (Jia et al., 2021; Zhang et al., 2022b). We analyze count-dependent frozen-feature scores under replacement of records and labels. Bounded-feature certificates and exact stored-source copy conditions address different intervention sets within this interface.

The role of counts depends on the learning objective. Permutation-invariant aggregation represents unordered sets (Zaheer et al., 2017), while sums and means retain different multiset information (Xu et al., 2019). Class-balanced losses and logit adjustment use frequency to address unequal training distributions (Cui et al., 2019; Menon et al., 2021). We fix initial coefficients rather than estimate a new class prior. Trusted-data learning supplies a separate comparison (Charikar et al., 2017): our protected subset fixes preprocessing and some anchors, and we explicitly compare against classifiers using that subset alone.

3 Source Replacement and Count Normalization

3.1 Frozen features and source permissions

A support set is a labeled multiset $D = C \cup U$, where C contains protected records and U contains replaceable records. The protected status belongs to a source position. Copying a protected example into a replaceable position does not grant the new record protected status. Let $v(x) \in \mathbb{R}^d$ denote the feature of an input, $u = v(x)$ the query feature, and t_c a fixed class anchor. All have norm at most one. The encoder, preprocessing, query, anchors, class vocabulary, and tie-breaking rule are fixed throughout an attack.

Each source is encoded independently of U. Generative text readouts are centered using C; native text embeddings use last-token readouts, and image models use projected features. Replacing a record cannot change untouched features or anchors. Appendix C specifies each family's pooling, centering, and normalization.

An attacker replaces at most m records in U, potentially changing both their inputs and labels. If class c initially has n_c records, with r_c removals and s_c insertions, then

$$n'_c = n_c - r_c + s_c, \qquad \sum_c r_c = \sum_c s_c \le m, \qquad 0 \le r_c \le |U_c|. \tag{1}$$

Every resulting class must be nonempty. These constraints follow from replacing records, rather than from a distributional assumption. Fixed-class-count replacement adds $r_c = s_c$ for every class. The main certificate allows the larger set in Equation 1.

3.2 How the retained count changes a score

Central trimming produces different retained counts at odd and even sample sizes. For sorted values $a_{(1)} \le \cdots \le a_{(n)}$, define

$$\tau(n) = \left\lfloor \frac{n-1}{2} \right\rfloor, \qquad L(n) = n - 2\tau(n), \qquad T(a) = \frac{1}{L(n)} \sum_{j=\tau(n)+1}^{n-\tau(n)} a_{(j)}. \tag{2}$$

Thus T is the usual median, averaging the two middle values when n is even, and $L(n)$ is one or two. For a source feature z_i with class c, set

$$p_i = 1 - u^\top z_i, \qquad q_i = 1 - z_i^\top t_c, \qquad d_c = 1 - u^\top t_c. \tag{3}$$

Cauchy–Schwarz places these quantities in $[0, 2]$. The two-term distance score considered here is

$$R_c = L(n_c)T(p_c) + \lambda d_c T(q_c), \qquad f_D(x) = \arg\min_c R_c, \qquad \lambda \ge 0. \tag{4}$$

LeFCert's Equation (11) retains $k - 2M$ values at a fixed shot count k and trimming parameter M (Lai et al., 2025). Equation 4 sets $M = \tau(n_c)$ and recomputes it after replacement. We call this variable-count extension the original rule throughout. Ties use fixed class index. Appendix A.3 gives the corresponding algebra for any fixed symmetric trimming policy.

One correct-label copy can change f_D even when each class retains the same feature distribution. Let $u = (1, 0)$, $z_0 = (0.6, 0.8)$, $z_1 = (0.5, \sqrt{3}/2)$, $z_2 = -z_0$, and $z_3 = -z_1$. Each class has 20 copies of z_c, with $t_c = z_c$. Then $p = (0.4, 0.5, 1.6, 1.5)$ and $q = 0$. The original scores are $(0.8, 1.0, 3.2, 3.0)$, so class 0 wins. Replacing one class-2 source with a class-1 copy changes them to $(0.8, 0.5, 1.6, 3.0)$, so class 1 wins. Equal protected copies have zero mean, making the construction compatible with protected-mean centering. This example isolates classwise sum scaling. Appendix A.5 gives a fixed-anchor construction in which both terms are positive and the same flip occurs.

3.3 Fixing count-dependent coefficients

Count-dependent coefficients give a systematic target for normalization. Write $R_c(D, x) = \mathbf{w}_c(n_c)^\top \mathbf{z}_c(D, x)$ for component statistics $\mathbf{z}_c$ and their coefficients $\mathbf{w}_c$. In Equation 4, these are $(T(p_c), d_c T(q_c))$ and $(L(n_c), \lambda)$. Let D_0 have initial counts $n_{0,c}$. Holding its statistics fixed, a count change induces

$$\chi_c = \left[\mathbf{w}_c(n'_c) - \mathbf{w}_c(n_{0,c})\right]^\top \mathbf{z}_c(D_0, x). \tag{5}$$

For a unique minimum-cost winner w and initial gap $g_{wc} = R_c - R_w > 0$, the changed gap is $g_{wc} - \chi_w + \chi_c$. Thus count variation reverses this comparison exactly when $\chi_w - \chi_c > g_{wc}$; equality invokes the fixed tie rule. The full coefficient vector matters, since even equal within-class coefficient ratios can hide changing classwise scales.

The correction fixes every explicit coefficient at the trustworthy initial support design D_0:

$$\widehat{R}_c(D, x) = \mathbf{w}_c(n_{0,c})^\top \mathbf{z}_c(D, x). \tag{6}$$

It preserves all initial scores, including ties, for arbitrary positive initial counts. It removes coefficient changes without assuming the statistics are invariant. The initial counts are never refreshed after replacement. Appendix I gives finite-count sensitivity, an exact coefficient/statistic decomposition, and the correction's scope.

3.4 The trimmed-distance specialization

For a common initial retained count L_*, Equation 6 can be divided by this positive constant. The resulting rule is

$$\widetilde{R}_c = T(p_c) + \frac{\lambda}{L_*} d_c T(q_c), \qquad \widetilde{f}_D(x) = \arg\min_c \widetilde{R}_c. \tag{7}$$

L_* remains fixed after replacement. Both evaluated designs use even initial class sizes, hence $L_* = 2$; normalization introduces no fitted coefficient.

Proposition 1 (Clean prediction preservation)**.** *If $L(n_c) = L_*$ for every class, then $R_c = L_* \widetilde{R}_c$ for every query and class. Therefore $f_D = \widetilde{f}_D$, including the tie-breaking order. The identity also holds after every replacement that preserves those class counts.*

The identity follows by factoring out $L_* > 0$ and applies to unequal all-even class sizes. In the construction above, the normalized score remains $(0.4, 0.5, 1.6, 1.5)$. For mixed parity, Equation 6 instead retains each class's own initial coefficient. Copies with different content can still move central order statistics.

3.5 Preserving the class mass of an exponential cache

Cache adaptation exposes the same count factor without trimming. For normalized support keys z_i, Tip-Adapter uses $k_i = \exp[-\beta(1 - u^\top z_i)]$ with kernel sharpness $\beta > 0$ (Zhang et al., 2022a). With cache weight $\alpha > 0$, fixed initial counts $n_{0,c}$, and current counts n_c, compare

$$S_c = 100 u^\top t_c + \alpha \sum_{i: y_i = c} k_i, \qquad \widetilde{S}_c = 100 u^\top t_c + \alpha \frac{n_{0,c}}{n_c} \sum_{i: y_i = c} k_i. \tag{8}$$

Both scores select the largest value and coincide initially. Writing the sum as n_c times the class mean makes Equation 8 an instance of Equation 6, with the text term unchanged. Bounded kernels $k_i \in [e^{-2\beta}, 1]$ give the source-level intervals in Appendix H.3.

3.6 When a single stored-source copy changes the decision

The cache permits an exact condition for one cross-class copy from the original support set. Assume all sources are mutable and initial counts exceed one. Let μ_c, h_c, l_c be the initial class kernel mean, maximum, and minimum for the query. For winner w, define the common initial gap $g_{wc} = S_w - S_c$. The following quantities measure four possible margin losses; here n_c denotes the initial count:

$$A_w = \frac{\alpha n_w}{n_w - 1}(h_w - \mu_w), \qquad B_c = \frac{\alpha n_c}{n_c + 1}(h_c - \mu_c),$$
$$C_c = \frac{\alpha n_c}{n_c - 1}(\mu_c - l_c), \qquad D_w = \frac{\alpha n_w}{n_w + 1}(\mu_w - l_w). \tag{9}$$

Theorem 1 (Exact single-copy margins)**.** *Over at most one permitted copy, including no change, the minimum winner-versus-rival gaps are*

$$g_{wc} - \alpha(h_w + h_c) \quad \textit{and} \quad g_{wc} - \max\{A_w + B_c, C_c + D_w\} \tag{10}$$

for the original and normalized caches, respectively.

The first normalized term removes a large winner kernel and inserts a large rival kernel. The second removes a small rival kernel and inserts a small winner kernel. Both can be worst; other copies are bounded by these sums. The prediction survives exactly when all rival minimum gaps are nonnegative and zero gaps favor w under the fixed tie rule. Appendix J proves attainability and extends the result to protected sources.

Normalization changes sensitivity from kernel magnitude to within-class deviation. Moreover, $A_w, D_w \le \alpha h_w$ and $B_c, C_c \le \alpha h_c$, so the normalized minimum gap is at least the original one. Every prediction stable under the original complete single-copy family remains stable after normalization. Constant class kernels give zero loss. Appendix J proves this inclusion and explains why protected-source and larger threats require separate analysis.

4 Certifying the Complete Modified Score

4.1 Source-level intervals

For fixed removal and insertion counts, order statistics provide explicit bounds. Consider one class and one scalar readout array $a \in [0, 2]$. Keep its protected values. Delete the r largest mutable values and insert s zeros to obtain a lower array; delete the r smallest mutable values and insert s twos to obtain an upper array. Let their central means be $T_a^-(r, s)$ and $T_a^+(r, s)$. Both use the actual resulting length $n_c - r + s$.

Lemma 1 (Replacement intervals)**.** *For every legal replacement profile (r, s) in class c, its modified score lies between*

$$\ell_c(r, s) = T_p^-(r, s) + \frac{\lambda d_c}{L_*} T_q^-(r, s),$$
$$h_c(r, s) = T_p^+(r, s) + \frac{\lambda d_c}{L_*} T_q^+(r, s). \tag{11}$$

Sorted values and their central average are coordinatewise monotone, so these deletion and insertion extrema bound every legal array. The nonnegative coefficient $\lambda d_c / L_*$ preserves the bounds. The two readouts share a source feature; separate extrema remain valid but need not be jointly attainable. Appendix A proves this sufficient bound and illustrates its conservatism.

4.2 Shared-budget certification

A prediction is certified when every competitor remains more costly than the current winner. Let $w = \widetilde{f}_D(x)$ and let $\mathcal{A}_{wc}(m)$ contain the pair profiles (r_w, s_w, r_c, s_c) that extend to Equation 1. Define

$$\Delta_m = \min_{c \neq w} \min_{\mathcal{A}_{wc}(m)} \left\{ \ell_c(r_c, s_c) - h_w(r_w, s_w) \right\}. \tag{12}$$

Theorem 2 (Prediction certificate)**.** *If $\Delta_m > 0$, every legal set of at most m source replacements preserves $\widetilde{f}_D(x)$.*

Proof. Any legal attack induces an enumerated profile. Lemma 1 places each competitor above its lower bound and the winner below its upper bound. Equation 12 therefore implies a strictly positive actual competitor-minus-winner score difference for every class. □

The evaluated supports allow efficient pairwise enumeration. Outside classes can balance every pair profile satisfying $r_w + r_c \leq m$ and $s_w + s_c \leq m$, without emptying a class. Prefix minima over rival budgets then compute the certificate in $O(Km^2)$ work per query after interval construction, for K classes. Appendix B gives the algorithm and capacity conditions. All scores reuse the same frozen features.

Fixed-count certification isolates the count-related certificate gap. Let $\mathcal{C}_{\text{old}}$, $\mathcal{C}_{\text{norm}}$, and $\mathcal{C}_{\text{fix}}$ be the certified-correct sets for the original, normalized, and common fixed-count rules. Restricting feasible profiles gives $\mathcal{C}_{\text{old}}, \mathcal{C}_{\text{norm}} \subseteq \mathcal{C}_{\text{fix}}$. Hence $H_m = |\mathcal{C}_{\text{fix}}| - |\mathcal{C}_{\text{old}}|$ bounds the possible increase under this interval construction. Appendix A proves the bound. It does not imply that every old certificate survives normalization.

5 Experiments

5.1 Paired evaluation on fresh samples

The evaluation measures clean-prediction preservation, certified accuracy, and executable attack harm on the same frozen readouts. The text panel uses Qwen2.5-0.5B-Instruct, Qwen3-0.6B, Llama-3.1-8B-Instruct, Gemma2-2B-it, and Qwen3-Embedding-0.6B (Qwen Team, 2024; 2025; Llama Team, 2024; Gemma Team, 2024; Zhang et al., 2025). Its tasks are AG News, TweetEval emotion, and a four-intent Banking77 subtask (Zhang et al., 2015; Barbieri et al., 2020; Casanueva et al., 2020). Each task has two support slices, each with four protected and sixteen mutable examples per class and eight test queries per class. The five models share 192 distinct test queries, giving 960 query-model pairs.

The image panel evaluates native CLIP-B/32, CLIP-B/16, and SigLIP-B/16 features on CIFAR-FS and EuroSAT (Radford et al., 2021; Zhai et al., 2023; Bertinetto et al., 2019; Helber et al., 2019). Each task has ten five-way episodes, with ten support examples and one query per class. The three encoders share 100 distinct image queries, giving 300 query-model pairs. We evaluate both zero and four protected examples per class, denoted C0 and C4, on the same images. The primary image condition is C0. Appendix D specifies the new-source selection, including the new Banking77 intents, and keeps earlier development samples separate.

The scoring rule and analysis were fixed before extracting the fresh features. We retain $\lambda = 25$, $L_* = 2$, and a primary replacement budget of four, and report the full budget curve from zero to four. Certified accuracy is the fraction of predictions that are both correct and certified. The executable attack replaces one mutable record with an existing example from another class, preserving the donor's correct label. We exhaustively enumerate this copy family separately for each query and method. Its harm count is the number of initially correct predictions that at least one such copy flips. It is a restricted attack at a smaller budget than the certificate, so the two measurements answer complementary questions.

5.2 Robustness with unchanged clean predictions

Normalization increases certification across the text panel and reduces copy harm across all eight encoders. Table 1 gives the paired primary results. Text certified accuracy rises from $408/960$ to $444/960$, a gain of 3.75 percentage points. All five text encoders improve. Image certified accuracy rises from $122/300$ to $124/300$, with gains for CLIP-B/32 and SigLIP and a tie for CLIP-B/16. The clean-correct totals remain $595/960$ for text and $203/300$ for images. Every individual clean prediction, not just the aggregate accuracy, is unchanged.

Table 1: Clean predictions are preserved while certification and copy resistance improve. Counts use the fresh confirmation panel. Certificates use four replacements; harm counts initially correct predictions flipped by one cross-class copy. Old/new denotes the original/normalized distance score. Each text row shares 192 queries, and each image row shares 100.

Encoder	N	Clean	Certified old/new	Harm old/new
Qwen2.5-0.5B	192	98	66/70	7/4
Qwen3-0.6B	192	93	47/49	19/10
Llama-3.1-8B	192	134	93/103	20/8
Gemma2-2B	192	135	108/111	11/2
Qwen3-Embedding	192	135	94/111	23/6
CLIP-B/32	100	66	39/40	9/5
CLIP-B/16	100	69	36/36	14/9
SigLIP-B/16	100	68	47/48	8/4

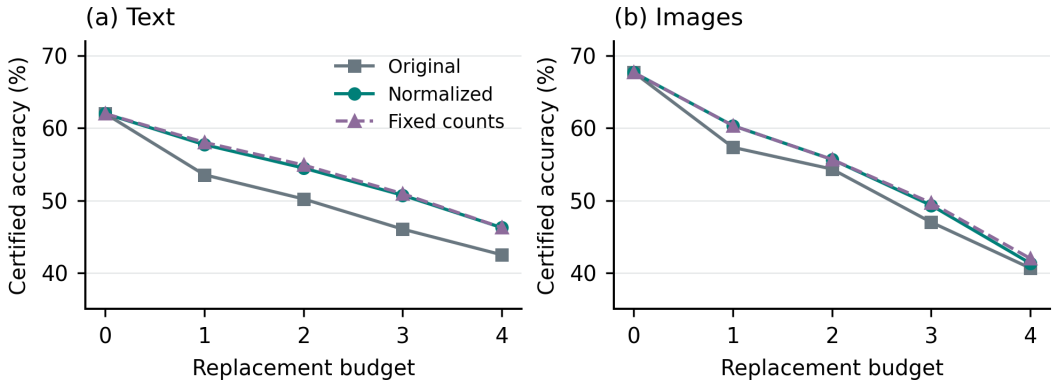


Figure 2: Count normalization narrows the gap to fixed-count certification. The curves report certified accuracy on fresh text and image queries as the allowed replacement budget increases. Fixed-count certification uses the same scalar intervals with count-preserving profiles. It is an explanatory control with a smaller threat set.

The benefit extends across the evaluated budget curve (Figure 2). Normalization adds 40 text and nine image certified-correct predictions at budget one, and 36 and two at budget four. No certified-correct prediction is lost at the primary budget. This empirical inclusion is distinct from Proposition 2's certificate-family bound.

Executable attacks show a larger harm reduction than the image certificate increment alone suggests. Text harm decreases from 80 to 30 query-model pairs, a 62.5% relative reduction; image harm decreases from 31 to 18, a 41.9% reduction. Every encoder improves, including CLIP-B/16, whose budget-four certified count is unchanged. Appendix Figure 4 gives the per-encoder counts. No saved copy witness flips a certified prediction. The attack measures finite-family susceptibility, while the certificate covers the broader replacement set.

Paired uncertainty is computed by task-stratified resampling of slices or episodes, keeping encoders paired. With 20,000 bootstrap replicates, the 95% intervals for certificate gains are $[3.54, 3.96]$ points for text and $[0.00, 1.67]$ for images. Six text clusters and twenty image clusters make these conditional descriptive intervals. The image interval includes zero, so the prespecified stronger criterion of positive lower bounds on every primary gain is not met. Appendix G reports all intervals. C4 repeats the same images and gives the same primary distance counts.

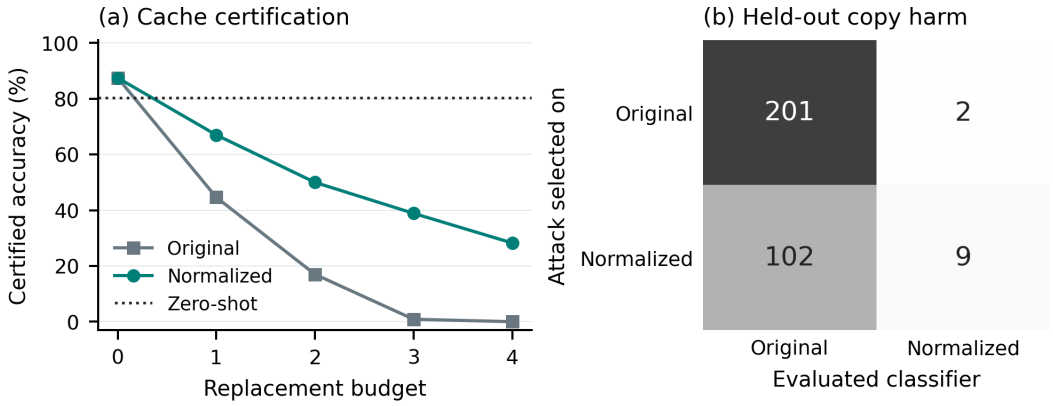


Figure 3: Cache normalization improves certification and shared-copy resistance in C0. Left: certified accuracy over 800 query-model pairs with identical clean predictions. Right: terminal shared-copy harm at budget four over 400 held-out pairs, by attack target and evaluated classifier. The 206/9 totals take maxima within each episode-encoder before summing, rather than maxima of these pooled columns.

5.3 The fixed-count certificate gap

Normalization closes the text panel's fixed-count certificate gap at budget four: $H_4 = 444 - 408 = 36$, and all 36 are recovered. Images recover two of four available certificates ($H_4 = 126 - 122$). Appendix Figure 5 gives the per-encoder decomposition.

The two remaining image certificates locate a residual effect beyond explicit scale. Asymmetric transfers increase the winner's upper bound more than the rival's lower bound (Appendix F.3). Changing counts can still change the central-statistic bounds, so positive H_4 permits a gain without ensuring it.

Preserving a predictor differs from selecting a new one. Normalized distance's 444 text certificates trail median distance's 497 and the C4 protected prototype's 682; its 124 image certificates trail zero-shot's 226. Table 9 gives the full comparison.

5.4 Cache accuracy and poisoning resistance

The cache study tests whether normalization preserves a useful adaptation gain. We evaluate training-free Tip-Adapter with ten augmented views per support image, using CLIP-B/32 and CLIP-B/16 on CIFAR-FS and EuroSAT. Each task has 20 new five-way, ten-shot support sets and two queries per class. The encoders share 400 distinct queries, giving 800 query-model pairs. We fix $\alpha = 2$ and $\beta = 1$ before inference and evaluate all sources as mutable. Appendix H supplies the complete protocol and protected-source condition.

Normalization preserves all 700 correct predictions, compared with 643 for zero-shot prediction. At four replacements, certified-correct counts rise from 0 to 226; one-record copy harm falls from 325 to 17, a 94.8% reduction. Every encoder-task combination has positive certificate and copy-resistance gains. Figure 3 shows the certificate curve and shared-attack results. These gains retain the cache's clean accuracy of 87.5%, including its 13.8-point advantage over zero-shot on EuroSAT.

Normalization also reduces shared-cache harm on queries withheld from path construction. For each support set, a greedy attack uses one query per class to select up to four record copies; the second query is reserved for evaluation. We generate attacks against each classifier and cross-evaluate both. Taking the more harmful terminal attack per episode, encoder, and evaluated classifier gives 206 versus 9 harmed held-out pairs. The paired improvements in certified accuracy, single-copy harm, and held-out shared harm are 28.25, 38.50, and 49.25 percentage points. Their simultaneous 95% cluster-bootstrap intervals are $[24.00, 32.50]$, $[34.38, 42.63]$, and $[44.00, 54.75]$, respectively. The held-out metric uses 400 pairs; the first two use 800. Appendix H.5 also reports all saved attack prefixes.

Table 2: Correct under every single cross-class stored-source copy in C0. Each task has 400 query-model pairs. This post hoc comparison reuses the cache study; it is distinct from the arbitrary-feature four-replacement certificate.

Task	Original cache	Normalized cache	Prototype	Zero-shot
CIFAR-FS	318	392	373	391
EuroSAT	57	291	264	252
Total	375	683	637	643

5.5 Adaptation gains under single-copy attacks

The exact condition lets us compare classifiers under the same copy threat on saved cache features. Table 2 counts correct predictions surviving every single cross-class copy, including no change. Full-support prototypes use normalized class-mean keys. All classifiers share the original donor pool and exhaustive enumeration, which agrees with Theorem 1's cache margins.

Normalization retains useful adaptation within this family. Its 683 robust-correct predictions exceed zero-shot by 40 and prototypes by 46. Of 67 cache-correct predictions that zero-shot gets wrong, 56 survive every single copy. The broader arbitrary-feature certificate covers 12 of these additions at budget one and none at budgets two to four.

Count sensitivity also predicts executable copies. Fixed-mean count changes reverse 315 of the 325 copy-vulnerable correct predictions in C0. Since maxima exceed means, Theorem 1 implies a realizable copy for each reversal. Appendix I gives the decomposition; Appendix K reports mixed-count checks and post hoc uncertainty. No new supports or encoder inference are added.

6 Discussion

Source permissions determine which aggregation properties require protection. Prior certificates cover different learning stages (Steinhardt et al., 2017; Levine & Feizi, 2021; Wang et al., 2024; Lai et al., 2025). In frozen-feature adaptation, preserving total support size need not preserve score coefficients. Fixing their initial values removes this dependence while retaining clean predictions.

Prediction preservation retains useful adaptation within a specified threat. Prototypes and class-text comparisons are effective alternatives (Snell et al., 2017; Radford et al., 2021; Zhai et al., 2023). Normalized Tip-Adapter (Zhang et al., 2022a) exceeds both under single copying (Table 2); zero-shot certifies more predictions under arbitrary four-record replacement. The source operation determines which benefit holds.

Count normalization differs from frequency modeling. Sum and mean encode different multiset information (Xu et al., 2019); class-balanced losses and logit adjustment use frequency to change decisions (Cui et al., 2019; Menon et al., 2021). We retain the initial predictor's coefficients. Within-class variation can still defeat a margin, so complete-score certification remains necessary after normalization.

The empirical scope remains the sampled supports and frozen interfaces. The trimmed-score image effect is small, and six text clusters limit population inference. The cache study adds forty supports and held-out shared attacks. Poisoning before encoding or through training dynamics (Shafahi et al., 2018; Geiping et al., 2021) requires its own dependency analysis.

7 Conclusion

Fixing initial coefficients preserves clean predictions while reducing poisoning sensitivity. Exact cache margins show that normalization cannot reduce robustness to a single cross-class copy when all sources are mutable. We also derive sufficient certificates for broader replacements. Within the single-copy family, the evaluated cache retains more correct predictions than zero-shot classification.

AI USE STATEMENT

Generative AI tools assisted mathematical exploration and proof development, hypothesis and experiment design, source-data organization, experiment implementation, interpretation of results, manuscript drafting, figure generation and plotting-code development, and literature search and organization. Numerical claims were checked against saved experiment records, and citations were checked against their primary sources. The authors are responsible for verifying the final text, proofs, code, and disclosures before submission.

ETHICS STATEMENT

The experiments study bounded changes to public benchmark support sets in an offline classification setting. The executable attack copies existing examples and retains their labels. No deployed service or third-party system was attacked. The findings concern the design and evaluation of defenses against support-set poisoning.

REPRODUCIBILITY STATEMENT

Appendices A and B provide the trimmed-score proofs and certification procedure. Appendices C and D specify model interfaces and data. Appendices F and G report complete distance-study results. Appendix H gives the cache protocol, certificates, and shared attacks. Appendices I and J develop the general correction and exact copy margins; Appendix K reports the reused-feature analysis. The accompanying single-cell code downloads the public inputs, computes model features, and produces the reported analyses, figures, and tables. The complete reproduction code and experimental results are provided in the Supplementary Material.

REFERENCES

Francesco Barbieri, Jose Camacho-Collados, Leonardo Neves, and Luis Espinosa-Anke. TweetEval: Unified Benchmark and Comparative Evaluation for Tweet Classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1644–1650. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.findings-emnlp.148. URL https://aclanthology.org/2020.findings-emnlp.148/.

Luca Bertinetto, João F. Henriques, Philip H. S. Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1805.08136.

Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning Attacks against Support Vector Machines. In *Proceedings of the 29th International Conference on Machine Learning*, 2012. URL https://arxiv.org/abs/1206.6389.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. Efficient Intent Detection with Dual Sentence Encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pp. 38–45. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.nlp4convai-1.5. URL https://aclanthology.org/2020.nlp4convai-1.5/.

Moses Charikar, Jacob Steinhardt, and Gregory Valiant. Learning from Untrusted Data. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 47–60. ACM, 2017. doi: 10.1145/3055399.3055491. URL https://arxiv.org/abs/1611.02315.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-Balanced Loss Based on Effective Number of Samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9268–9277, 2019. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Cui_Class-Balanced_Loss_Based_on_Effective_Number_of_Samples_CVPR_2019_paper.html.

Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. CLIP-Adapter: Better Vision-Language Models with Feature Adapters. *International Journal of Computer Vision*, 132(2):581–595, 2024. doi: 10.1007/s11263-023-01891-x. URL https://link.springer.com/article/10.1007/s11263-023-01891-x.

Jonas Geiping, Liam Fowl, W. Ronny Huang, Wojciech Czaja, Gavin Taylor, Michael Moeller, and Tom Goldstein. Witches' Brew: Industrial Scale Data Poisoning via Gradient Matching. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2009.02276.

Gemma Team. Gemma 2: Improving Open Language Models at a Practical Size, 2024. URL https://arxiv.org/abs/2408.00118.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019. doi: 10.1109/jstars.2019.2918242. URL https://arxiv.org/abs/1709.00029.

Jinyuan Jia, Xiaoyu Cao, and Neil Zhenqiang Gong. Intrinsic Certified Robustness of Bagging against Data Poisoning Attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 7961–7969, 2021. doi: 10.1609/aaai.v35i9.16971. URL https://ojs.aaai.org/index.php/AAAI/article/view/16971.

Yuni Lai, Xiaoyu Xue, Linghui Shen, Yulun Wu, Gaolei Li, Song Guo, Kai Zhou, and Bin Xiao. Provably Robust Adaptation for Language-Empowered Foundation Models, 2025. URL https://arxiv.org/abs/2510.08659.

Alexander Levine and Soheil Feizi. Deep Partition Aggregation: Provable Defenses against General Poisoning Attacks. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2006.14768.

Llama Team. The Llama 3 Herd of Models, 2024. URL https://arxiv.org/abs/2407.21783.

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2007.07314.

Saif M. Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. SemEval-2018 Task 1: Affect in Tweets. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pp. 1–17. Association for Computational Linguistics, 2018. doi: 10.18653/v1/S18-1001. URL https://aclanthology.org/S18-1001/.

Qwen Team. Qwen2.5 Technical Report, 2024. URL https://arxiv.org/abs/2412.15115v1.

Qwen Team. Qwen3 Technical Report, 2025. URL https://arxiv.org/abs/2505.09388.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Elan Rosenfeld, Ezra Winston, Pradeep Ravikumar, and J. Zico Kolter. Certified Robustness to Label-Flipping Attacks via Randomized Smoothing. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8230–8241. PMLR, 2020. URL https://proceedings.mlr.press/v119/rosenfeld20b.html.

Ali Shafahi, W. Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/hash/22722a343513ed45f14905eb07621686-Abstract.html.

Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical Networks for Few-shot Learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/cb8da6767461f2812ae4290eac7cbc42-Abstract.html.

Jacob Steinhardt, Pang Wei Koh, and Percy Liang. Certified Defenses for Data Poisoning Attacks. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/9d7311ba459f9e45ed746755a32dcd11-Abstract.html.

Yanting Wang, Wei Zou, and Jinyuan Jia. FCert: Certifiably Robust Few-Shot Classification in the Era of Foundation Models. In *2024 IEEE Symposium on Security and Privacy (SP)*, pp. 2939–2957. IEEE, 2024. doi: 10.1109/sp54263.2024.00240. URL https://arxiv.org/abs/2404.08631.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How Powerful are Graph Neural Networks? In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1810.00826.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan Salakhutdinov, and Alexander Smola. Deep Sets. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/f22e4747da1aa27e363d86d40ff442fe-Abstract.html.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid Loss for Language Image Pre-Training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11975–11986, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/html/Zhai_Sigmoid_Loss_for_Language_Image_Pre-Training_ICCV_2023_paper.html.

Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-Adapter: Training-free Adaption of CLIP for Few-shot Classification. In *Computer Vision – ECCV 2022*, volume 13695 of *Lecture Notes in Computer Science*, pp. 493–510. Springer, 2022a. doi: 10.1007/978-3-031-19833-5_29. URL https://arxiv.org/abs/2207.09519.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level Convolutional Networks for Text Classification. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/hash/250cf8b51c773f3f8dc8b4be867a9a02-Abstract.html.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models, 2025. URL https://arxiv.org/abs/2506.05176.

Yuhao Zhang, Aws Albarghouthi, and Loris D'Antoni. BagFlip: A Certified Defense Against Data Poisoning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 31474–31483. Curran Associates, Inc., 2022b. doi: 10.52202/068431-2282. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/cc19e4ffde5540ac3fcda240e6d975cb-Abstract-Conference.html.

APPENDIX GUIDE

The appendix follows the dependencies of the main argument. Section A proves the source-level bounds and distinguishes feature-space constructions from input-level attacks. Section B specifies the certificate computation. Sections C and D describe model interfaces and data. Section E defines comparison scores. Sections F and G provide distance results and uncertainty. Section H develops the exponential-cache certificates, sample protocol, shared attacks, and results. Sections I and J derive general count sensitivity and exact single-copy margins. Section K reports their reused-feature analysis and common-threat comparisons. Section L separates earlier evidence from the fresh study; Section M gives the reproduction order.

A PROOFS AND CONSTRUCTIONS

A.1 OBJECTS AND ASSUMPTIONS

The guarantee is conditional on a fixed feature computation and a declared set of editable sources. A query x is evaluated using an ordered class vocabulary $\{0, \dots, K-1\}$. Each source position stores one input and one label. The protected positions C are immutable; the remaining positions U can be replaced. A replacement changes the record stored at a position, including its label. Consequently, an inserted copy of a protected input remains mutable when placed in U. Source permissions are independent of feature equality or duplicate content.

The encoder and all shared preprocessing are fixed before replacement. The feature map can depend on the protected records, denoted v_C, provided those records remain unchanged. The query and class anchors also remain fixed. Thus every untouched source retains its feature after the attack. We require $\|u\|, \|z_i\|, \|t_c\| \le 1$. The finite-dimensional construction below uses unit vectors, whereas the implementation also permits zero vectors. The central mean is defined only for nonempty class arrays. Legal attacks therefore leave each class nonempty, and the evaluated budgets satisfy this property automatically.

The score has nonnegative distance weight λ, and $L_* > 0$ is fixed by the clean support design. The primary experiments use $\lambda = 25$ and $L_* = 2$. Source replacement is the only intervention considered by the certificate: it does not change the encoder, query, anchor, or preprocessing. These are interface assumptions, not properties inferred from a Transformer architecture.

A.2 PRESERVATION UNDER A COMMON RETAINED COUNT

The retained-count identity follows from integer arithmetic. If $n = 2k+1$, then $\tau(n) = k$ and $L(n) = 1$. If $n = 2k$, then $\tau(n) = k-1$ and $L(n) = 2$. Therefore

$$T(a) = \begin{cases} a_{(k+1)}, & n = 2k+1, \\ \big(a_{(k)} + a_{(k+1)}\big)/2, & n = 2k. \end{cases} \tag{13}$$

Both cases are the median convention used by the implementation. The central sum is exactly $L(n)T(a)$.

For a support design with $L(n_c) = L_*$ in every class, the original score satisfies

$$R_c = L_* T(p_c) + \lambda d_c T(q_c) \tag{14}$$

$$= L_* \left(T(p_c) + \frac{\lambda}{L_*} d_c T(q_c) \right) = L_* \widetilde{R}_c. \tag{15}$$

Hence $R_c - R_j = L_*(\widetilde{R}_c - \widetilde{R}_j)$ for every pair of classes. Strict comparisons and ties coincide. Applying a shared deterministic class-index rule proves equality of predictions. Equal class sizes are sufficient but unnecessary. In particular, sizes 10, 12, 20, and 24 can appear together while retaining $L_* = 2$.

The same proof applies pointwise after any attack that preserves every class count. Such an attack can change features, labels within the allowed profile, order statistics, and score magnitudes. None of these changes alters the common multiplier. This establishes a useful control: differences between the original and normalized predictors require profiles that leave the common-retained-count condition.

A.3 Extension to fixed symmetric trimming policies

The scale identity and interval proof extend beyond maximal trimming. Let $\tau_0(n)$ be any fixed integer-valued policy satisfying $0 \le 2\tau_0(n) < n$. Write $L_0(n) = n - 2\tau_0(n)$ and let T_0 average the values retained after removing $\tau_0(n)$ from each end. The corresponding original and normalized scores are

$$R_c^0 = L_0(n_c)T_0(p_c) + \lambda d_c T_0(q_c), \qquad \widetilde{R}_c^0 = T_0(p_c) + \frac{\lambda}{L_*} d_c T_0(q_c). \tag{16}$$

If the clean design has $L_0(n_c) = L_*$ for every class, then $R^0 = L_* \widetilde{R}^0$. At a fixed replacement profile, the resulting length and trimming ranks are fixed, and T_0 is still a nonnegative average of order statistics. The scalar extremal-array proof and the shared-budget certificate therefore apply unchanged. The fixed-count certificate-family inclusion follows by the same profile restriction.

This extension identifies the mathematical condition. A varying retained count can change score scale under any such policy, but neither a flip nor a robustness gain is guaranteed. The primary distance experiments use maximal trimming; Appendix K analyzes two other fixed policies on the saved features. FCert's retained-distance mean corresponds to the first normalized term, whereas LeFCert's Equation (11) combines a retained sum with a retained mean (Wang et al., 2024; Lai et al., 2025). The variable-count continuation and its fixed-design normalization are the objects analyzed here.

A.4 A centering-compatible count-only construction

The main construction realizes the required feature geometry explicitly. Use the two-dimensional vectors

$$u = (1, 0), \quad z_0 = (3/5, 4/5), \quad z_1 = (1/2, \sqrt{3}/2), \quad z_2 = -z_0, \quad z_3 = -z_1. \tag{17}$$

Set $t_c = z_c$, and give every class 20 copies of z_c. Four of the copies in each class can be protected. All vectors have norm one. Equal protected multiplicities imply

$$\frac{1}{16} \sum_{i \in C} z_i = \frac{1}{4}(z_0 + z_1 + z_2 + z_3) = 0. \tag{18}$$

Taking these vectors as raw protected readouts makes protected-mean centering leave them unchanged. The construction therefore satisfies both the normalized-feature assumptions and the algebraic centering constraint of the text interface.

For every class, the support-to-anchor distance is zero, while the query-to-support distance is constant within the class:

$$T(q_c) = 0, \qquad T(p_c) = a_c, \qquad a = (2/5, 1/2, 8/5, 3/2). \tag{19}$$

The original clean score is $2a$, whose unique minimizer is class 0. Replace one mutable class-2 record by a class-1 copy. Only n_1 and n_2 change, to 21 and 19. The score becomes $(4/5, 1/2, 8/5, 3)$, whose unique minimizer is class 1. Every class continues to have empirical feature distribution concentrated on its original z_c. The normalized score remains a, retaining class 0.

This construction demonstrates an attainable configuration in feature space. It does not claim that specified natural-language strings or images map exactly to these vectors under a particular pretrained encoder. The executable benchmark copies supply the separate input-level evidence. Keeping these roles separate makes the mechanism proof independent of a claim about encoder preimages.

A.5 A construction with two positive score terms

The same count change also flips a score whose two terms are both active. Keep the query, sources, labels, and protected positions of the preceding construction, but use fixed unit anchors $t_c = Az_c$, where

$$A = \frac{1}{401} \begin{pmatrix} 399 & -40 \\ 40 & 399 \end{pmatrix}. \tag{20}$$

Since $399^2 + 40^2 = 401^2$, A is a rotation. It gives $q_c = 1 - z_c^\top A z_c = 2/401$ in every class. For $\lambda = 25$, write $b = 50/401$ and $d_c = 1 - u^\top A z_c$. The original score is $2a_c + bd_c$ before copying and $L(n_c')a_c + bd_c$ afterward. Both summands are positive. The normalized score is $a_c + (b/2)d_c$ on both support sets.

The original winner changes from class 0 to class 1, while the normalized winner stays at class 0. In fact $d_1 > d_0$, so the original and normalized clean scores both prefer class 0 to class 1. The other classes have $a_c \geq 1.5$ and cannot win. After the class-2-to-class-1 replacement, the original class-0 minus class-1 score is

$$0.3 + b(d_0 - d_1) \geq 0.3 - 2b = 0.3 - 100/401 > 0. \tag{21}$$

The approximate clean original scores are $(0.8602, 1.0734, 3.3892, 3.1760)$; after copying they are $(0.8602, 0.5734, 1.7892, 3.1760)$. Normalized scores remain $(0.4301, 0.5367, 1.6946, 1.5880)$. This is a construction for independently fixed anchors, as in a class-text interface. It is distinct from requiring anchors to equal the means of identical protected sources. The zero-second-term example isolates sum scaling; this example verifies that a positive second term does not remove it.

A.6 Order-statistic bounds with protected values

Fix one scalar readout array and write its mutable values in ascending order as $b_1 \leq \cdots \leq b_M$. After deleting r mutable values, any retained mutable subsequence has length $M - r$. Its jth value has an original index between j and $j + r$, so

$$b_j \leq b_{i_j} \leq b_{j+r}, \qquad 1 \leq j \leq M - r. \tag{22}$$

The lower subsequence $(b_1, \ldots, b_{M-r})$ removes the largest values, and the upper subsequence $(b_{r+1}, \ldots, b_M)$ removes the smallest values. Equation 22 bounds every possible retained subsequence componentwise.

Merging an identical protected array preserves this sorted order. One way to see this is through threshold counts: if an array is componentwise smaller, it has at least as many elements below any threshold. Adding the same protected elements preserves that inequality, so the corresponding order statistics remain ordered. Inserting s zeros gives a lower bound for any s additional values in $[0, 2]$; inserting s twos gives an upper bound. Sorting the merged arrays therefore produces componentwise lower and upper bounds for every legal changed-class array of length $n_c - r + s$.

The central mean is coordinatewise monotone at this fixed resulting length. It averages either one or two specified order statistics with nonnegative coefficients. Applying it to the bounding arrays yields

$$T_a^-(r, s) \leq T(a') \leq T_a^+(r, s). \tag{23}$$

Apply this argument separately to p and q. Cauchy–Schwarz gives the shared bounded range, and $\lambda d_c / L_* \geq 0$ preserves the inequalities when combining terms. This proves Lemma 1.

The independent bounds deliberately discard part of the joint geometry. For unit query and anchor with $\rho = u^\top t$, let $\alpha = u^\top z = 1 - p$ and $\beta = t^\top z = 1 - q$. A vector with norm at most one can attain these two projections only if

$$\alpha^2 - 2\rho\alpha\beta + \beta^2 \leq 1 - \rho^2 \tag{24}$$

when $|\rho| < 1$. This follows by minimizing the squared norm of a vector with the two prescribed projections in the span of u and t. The inverse of their Gram matrix gives the left side divided by $1 - \rho^2$. If $\rho = 1$ or -1, the corresponding relation is $\beta = \alpha$ or $\beta = -\alpha$. Independent scalar endpoints can violate these relations, and optimal deletions for p and q can remove different records. The combined interval still contains every attainable score.

A.7 A stable prediction that the interval does not certify

A geometric example shows why the sufficient interval is not a necessary condition for robustness. Let $u = (1, 0)$, $L_* = 2$, and $\lambda = 25$. Class 0 has two protected copies of $z_0 = (-1/2, \sqrt{3}/2)$ with anchor $t_0 = u$. Its score is $3/2$ because $d_0 = 0$. Class 1 has two mutable copies of $-u$ and fixed anchor $t_1 = -u$, giving score 2. These anchors are part of the fixed-anchor interface for this example.

After a same-class replacement in class 1, let $p = 1 - u^\top z$ for the inserted feature. Since $t_1 = -u$, its other readout is $q = 2 - p$. The rival score is

$$\frac{2+p}{2} + 25\frac{2-p}{2} = 26 - 12p \geq 2. \tag{25}$$

A replacement from class 1 into class 0 leaves one original class-1 record with score 2. The median of two protected class-0 duplicates and one inserted value stays at $3/2$ in its query distance, and the second term still has zero coefficient. These exhaust the possible nontrivial single replacements for the two-class vocabulary. The winner therefore remains class 0 for every allowed feature.

The independent interval fails on the legal same-class profile $r_1 = s_1 = 1$. The rival query-distance lower mean is $(2 + 0)/2 = 1$, and its anchor-distance lower mean is zero. Their combination gives $\ell_1 = 1$, below the winner's score $3/2$. The endpoints $p = 0$ and $q = 0$ cannot arise together because $q = 2 - p$. This example isolates interval conservatism without contradicting the prediction certificate.

A.8 Pair profiles and global budget completion

A pairwise certificate must cover the profile induced by every global replacement. Let the winner and one rival have total removals $R = r_w + r_c$ and insertions $S = s_w + s_c$. Necessarily $R, S \leq m$. To complete the profile, an imbalance $S - R > 0$ requires $S - R$ further removals outside the pair. An imbalance $R - S > 0$ requires $R - S$ further insertions outside the pair. If the required removable capacity and a valid outside class exist, this uses exactly $\max(R, S)$ replacements.

The experimental panels meet these completion conditions for all enumerated profiles. Text has four classes with sixteen mutable records in each; images have five classes with ten mutable records in C0 and six in C4. The maximum budget is four. A third class therefore has enough removable records to balance either pair, and no class can be emptied. Under these conditions, pair-profile feasibility is exactly the rectangular constraints $R \leq m$ and $S \leq m$ together with nonnegative component counts.

For a general interface, the completion test is part of the legal profile definition. With two classes, there is no outside class to absorb an imbalance, so pair totals must agree. With limited mutable capacity, a requested extra removal may be impossible. Enumerating a superset of legal profiles still gives a sufficient certificate, but may reduce its coverage. It should not be described as the exact projection of the global attack set in that case.

A.9 Proof of shared-budget certification

Fix a legal attacked support D' and a competitor $c \neq w$. Its global source operation induces a pair profile in $\mathcal{A}_{wc}(m)$. By Lemma 1,

$$\widetilde{R}'_c - \widetilde{R}'_w \geq \ell_c(r_c, s_c) - h_w(r_w, s_w) \geq \Delta_m. \tag{26}$$

If $\Delta_m > 0$, the winner has a strictly smaller score than every competitor. Its identity is therefore unchanged. At a zero interval margin, the procedure withholds certification; a tie may be stable under the fixed tie rule, but strict positivity supplies a simple sufficient condition that is uniform across classes.

The two-dimensional prefix minimum is an exact computational identity for this interval program. Define

$$P_c(a, b) = \min_{\substack{0 \leq r \leq a,\ 0 \leq s \leq b \\ r \leq |U_c|,\ n_c - r + s \geq 1}} \ell_c(r, s), \qquad a, b \in \{0, \ldots, m\}, \tag{27}$$

where all indices are integers. For fixed winner counts (r_w, s_w), the rival can use any $r_c \leq m - r_w$ and $s_c \leq m - s_w$ in the evaluated panels. Its smallest lower bound is exactly $P_c(m - r_w, m - s_w)$. Minimizing this difference over winner counts and rivals recovers Equation 12. The identity removes redundant enumeration, not dependence on the declared attack budget.

A.10 Proof and interpretation of the fixed-count bound

The fixed-count interval program restricts each class to $r_c = s_c$. Its resulting lengths equal their initial values. The original and normalized class bounds are therefore related by L_*, just as their

Table 3: Certificate computation for the evaluated source capacities. Each step uses the same fixed feature readouts.

Step	Operation
1	Compute bounded source-query and source-anchor readouts and the fixed query-anchor distances.
2	For each class, sort mutable values separately for each readout and retain protected values.
3	For each $0 \le r, s \le M$, form the lower and upper retained arrays at length $n_c - r + s$, then evaluate their central means.
4	Combine the scalar bounds using the full score coefficients to obtain $\ell_c(r, s)$ and $h_c(r, s)$.
5	Find the clean winner from the zero-edit scores. For each rival, construct the two-dimensional prefix minimum of its lower bounds.
6	At each budget, minimize competitor lower minus winner upper over the remaining shared removal and insertion budgets.
7	Return the prediction and all margins; certify only margins strictly greater than zero.

scores are. Minima over the same profiles preserve that positive scaling. Hence their fixed-count certificate decisions coincide for each query at every budget.

Proposition 2 (A certificate-family bound)**.** *For an initial design satisfying Proposition 1,*

$$\mathcal{C}_{\mathrm{old}}, \mathcal{C}_{\mathrm{norm}} \subseteq \mathcal{C}_{\mathrm{fix}}, \qquad |\mathcal{C}_{\mathrm{norm}}| - |\mathcal{C}_{\mathrm{old}}| \le H_m := |\mathcal{C}_{\mathrm{fix}}| - |\mathcal{C}_{\mathrm{old}}|. \tag{28}$$

Variable-count certification must also pass all fixed-count profiles. Thus every certified prediction under either variable-count rule belongs to the common fixed-count certificate set. Since clean predictions coincide, intersecting with the ground-truth-correct set preserves both inclusions. Taking cardinalities proves Proposition 2; it also yields

$$|\mathcal{C}_{\mathrm{norm}} \setminus \mathcal{C}_{\mathrm{old}}| \le |\mathcal{C}_{\mathrm{fix}} \setminus \mathcal{C}_{\mathrm{old}}| = H_m. \tag{29}$$

The bound concerns this certificate family. A more informative joint-feature bound could enlarge the fixed-count set and change the reference gap.

The two variable-count scores do not have a universal dominance relation. When $L_* = 2$,

$$\widetilde{R}_c - \frac{R_c}{2} = \left(1 - \frac{L(n_c)}{2}\right) T(p_c). \tag{30}$$

This nonnegative increment occurs only in odd-size classes. Its effect on a prediction depends on whether the changed class is the winner or a competitor and on the remaining score differences. Consequently, $H_m = 0$ excludes positive certified-correct gain within the family but does not exclude a loss. A positive H_m permits, rather than ensures, a gain.

B Certification and Attack Algorithms

B.1 Inputs and output conventions

The certificate takes source features Z, source labels y, a protected-position mask, a query feature u, fixed class anchors, and a maximum replacement budget M. The classifier first evaluates the unmodified support set and selects its winner by minimum class index among tied minima. The output contains the clean prediction and a strict interval margin for every budget $m \in \{0, \ldots, M\}$. A positive margin means certified stability. Ground-truth query labels are used only afterward to compute certified accuracy.

B.2 Complexity and shared readouts

Feature extraction is independent of the certificate budget. Each unique source, query, and required class text is encoded once for a given encoder and sampling block. The original score, normalized

Table 4: CPU certificate-stage wall time for the complete primary panel, in seconds. Each pass jointly evaluates six scores, five budgets, and two threat variants. Ranges are the minimum and maximum of three passes, not confidence intervals.

Stage	Median	Range
Scalar intervals and score bounds	2.041	$[2.032, 2.060]$
Budget minimization	2.522	$[2.506, 2.524]$
Total within each pass	4.565	$[4.538, 4.582]$

score, reference scores, and attacks share these saved features. Sorting mutable scalar readouts takes $O(\sum_c n_c \log n_c)$ time per query, with source-anchor sorting reusable across queries. A direct implementation of each interval merges and sorts the retained arrays; more efficient order-statistic selection can reduce this work without changing the bounds.

After the class interval arrays are available, constructing a two-dimensional prefix minimum takes $O(KM^2)$ time and memory per query. Evaluating one budget takes $O(Km^2)$ winner-profile work. Computing all budgets up to M by direct repetition takes $O(KM^3)$ time. These costs describe the interval stage, not end-to-end latency. We do not infer a runtime advantage over other defenses from these asymptotic expressions.

B.3 Exhaustive one-record copies

The executable attack enumerates every ordered pair consisting of a removable source and a donor source with a different class label. For each pair, it replaces the removable record with the donor's feature and correct label, then recomputes every class score on the changed multiset. A donor can be protected, but its inserted copy occupies the removable source position and remains mutable. Every scoring rule is evaluated on each changed support set, using its own clean prediction as the comparison point.

The procedure records whether at least one copy flips each query's clean prediction. It also records a successful removal/donor pair when one exists and the predictions of all compared scores under that pair. Harm intersects the flip indicator with clean correctness. An already incorrect prediction becoming a different incorrect prediction is a flip but does not count as harmed correctness. This distinction keeps the attack metric aligned with its stated utility interpretation.

The copy family is finite and exhaustive within its definition. Text has $4 \cdot 3 \cdot 16 \cdot 20 = 3{,}840$ candidate copies per query. Images have $5 \cdot 4 \cdot 10 \cdot 10 = 2{,}000$ candidates in C0 and $5 \cdot 4 \cdot 6 \cdot 10 = 1{,}200$ in C4. These counts include source multiplicities. The operation is chosen separately for each query, so the total harm is not the outcome of a single poisoned support set shared by all queries. The certificate allows arbitrary replacement inputs and labels at budgets up to four, which strictly exceeds this attack family.

B.4 Measured certificate-stage cost

A post hoc CPU measurement quantifies the cost of the independent certificate implementation. The complete primary panel contains 90 units and 1,260 query-model pairs: 30 text C4 units and 60 image C0 units. The measurement computes all six source-dependent scores together, all five budgets, and both fixed-count and variable-count margins. It uses direct feasible-pair enumeration rather than the production prefix-minimum solver.

One full warmup precedes three timed passes. Computation uses float64, Python 3.13.15, NumPy 2.1.3, and one BLAS thread. The result archive's `analysis/binding_and_timing.json` records every pass, and `environment/` records package versions. Wall time covers scalar interval construction and budget minimization. File access, feature preparation, dot products, clean prediction, serialization, encoder inference, and attack enumeration are excluded. CPU affinity and background load are not controlled; memory is not measured.

The total median is computed from pass totals and need not equal the sum of stage medians. It corresponds to 3.62 milliseconds per query-model pair amortized over the six jointly evaluated methods.

It is neither a single-method latency nor an old-versus-new speedup measurement. The timing procedure is included in the reproduction code, and the result archive retains its pass records.

C Model Interfaces and Fixed Dependencies

C.1 Generative text encoders

The four generative models are used as frozen feature encoders, not as answer generators. The input contains a task description, the four class names with codes A to D, and the text being classified. The model's native chat template is applied with an assistant-generation prefix. The hidden representation at the last nonpadding position is read without applying a language-model output head. Qwen3 uses the nonthinking template option. No sampled token sequence participates in classification.

The user message has the following fixed form, with the task, class names, and source text substituted:

```
Classify the [task]. Codes: A=[class 0]; B=[class
1]; C=[class 2]; D=[class 3]. Return only one code.
Text: [text]
```

Underscores in class names are replaced with spaces. The model-specific chat template supplies the surrounding special tokens. Padding is on the left, and the last hidden position is therefore the last nonpadding position. Inputs are not truncated. The tokenizer's end-of-sequence token serves as padding when no separate padding token is defined.

Protected-mean centering makes the feature transform independent of mutable sources. If $h(x)$ is the raw readout, define

$$\mu_C = \frac{1}{|C|} \sum_{i \in C} h(x_i), \qquad v_C(x) = \text{normalize}(h(x) - \mu_C), \tag{31}$$

$$t_c = \text{normalize}\left(\frac{1}{|C_c|} \sum_{i \in C_c} v_C(x_i) \right). \tag{32}$$

The protected samples are the same four records per class for the original and normalized rules. Their means and anchors remain fixed under all allowed replacements. The full input passes through each model's actual attention, normalization, position encoding, and feed-forward blocks.

Each encoder retains its native attention computation, including its normalization, position encoding, and attention-logit transformations. Disabling generation caching only avoids retaining keys and values for later generation steps. The source-local guarantee follows from independent encoding and fixed preprocessing, without requiring the models to share an attention formula.

C.2 Native text and image embeddings

Qwen3-Embedding uses the last-token embedding and normalization without protected-mean centering (Zhang et al., 2025). Sample inputs use the instruction `Instruct: Given a text, identify its [task].` followed by a newline and `Query:[text]`. Class anchors are the embeddings of the corresponding plain class names. Both queries and support texts use the same task instruction. The class-text-only reference compares query features directly with these anchors.

CLIP and SigLIP use their native image and text projection interfaces (Radford et al., 2021; Zhai et al., 2023). Images pass through the corresponding model processor before feature normalization. Class text uses `A photo of a [class]` for CIFAR-FS and `a centered satellite photo of [class].` for EuroSAT. Image anchors are fixed text features in both C0 and C4. The protected-prototype reference in C4 instead forms class representatives from the four protected image features. It does not alter the anchors of the distance adapter.

The zero-shot comparison uses normalized image-text inner products. A positive shared temperature and a class-independent bias preserve their class ordering. Native projected features are therefore sufficient for this comparison. The theoretical bounded range follows from normalization, while the empirical classification quality depends on the particular pretrained features and templates.

Table 5: Model identifiers used for the fresh confirmation study. Llama and Gemma use the stated distribution mirrors; the identifiers distinguish the actual tested weights from other releases.

Encoder	Model identifier
Qwen2.5	`Qwen/Qwen2.5-0.5B-Instruct`
Qwen3	`Qwen/Qwen3-0.6B`
Llama 3.1	`unsloth/Meta-Llama-3.1-8B-Instruct`
Gemma 2	`unsloth/gemma-2-2b-it`
Qwen3-Embedding	`Qwen/Qwen3-Embedding-0.6B`
CLIP-B/32	`openai/clip-vit-base-patch32`
CLIP-B/16	`openai/clip-vit-base-patch16`
SigLIP-B/16	`google/siglip-base-patch16-224`

C.3 Precision and execution

Text weights and inference use bfloat16. The vision models retain float32 weights and use bfloat16 autocasting for forward computation. Raw text and image readouts are saved in float32. Cache support images retain one readout per augmented view; averaging those readouts and normalizing the resulting keys use float64. Feature normalization, scalar scores, and certificate intervals also use float64. Scalar distances are clipped to $[0, 2]$ to remove numerical excursions at the bounded-range endpoints. A zero vector remains zero under normalization; the denominator is bounded below by 10^{-30}.

The mathematical results concern real-valued bounds. Numerical agreement of independent implementations does not itself supply directed-rounding guarantees for every floating-point operation. The reported certificates are the strict-margin decisions of the specified float64 interval program. Saved readouts and margins make those decisions reproducible. There are no trainable adapter parameters in the primary rule and no backbone updates. Batch size changes only inference throughput, not the intended source selection or score definition.

D Datasets and Independent Source Selection

D.1 Text tasks and support slices

The text tasks cover topic, emotion, and intent classification. AG News uses its four topic labels: World, Sports, Business, and Sci/Tech (Zhang et al., 2015). TweetEval uses a four-class, single-label emotion configuration with anger, joy, optimism, and sadness (Barbieri et al., 2020), derived from the emotion-classification data of SemEval-2018 Task 1 (Mohammad et al., 2018). Banking77 uses four fixed intents in this study: age limit, cash withdrawal charge, lost or stolen phone, and transfer into account (Casanueva et al., 2020). The Banking77 results concern this four-intent task, not a full 77-class evaluation.

Every task supplies two nonoverlapping slices. Each slice contains four protected training records per class, sixteen mutable training records per class, and eight test records per class. It therefore contains 80 support records and 32 queries. Across the six slices, there are 480 support records and 192 test queries, or 672 distinct texts. Five encoders share those texts. Native class-text anchors add separate encoding requests but do not add test examples.

The fresh source selection excludes the registered development texts before any model readout. We remove 1,825 previously used stripped text contents, then sample the new sources using the fixed selection procedure. No exact stripped-text overlap remains between these registered prior sources and the new 672 texts. The original four Banking77 development intents lacked enough remaining test examples for two new slices. We therefore selected four new intents once, without consulting model outcomes. This is confirmation on new intents within the dataset, rather than replication on the identical four-label task. Exact source identifiers and their roles are retained in the selection record.

Table 6: Public dataset distributions and task scope. Text support records come from training splits and queries from test splits. Image support and query indices are distinct within the frozen episode construction.

Task	Distribution identifier	Classes
AG News	`fancyzhx/ag_news`	4
TweetEval emotion	`cardiffnlp/tweet_eval`, emotion	4
Banking77 subtask	`mteb/banking77`	4
CIFAR-FS	`uoft-cs/cifar100`, evaluation classes	20
EuroSAT	`timm/eurosat-rgb`	10

D.2 Image tasks and episodes

The image tasks use the CIFAR-FS evaluation class set and EuroSAT RGB images (Bertinetto et al., 2019; Helber et al., 2019). CIFAR-FS is constructed from the CIFAR-100 distribution using its twenty evaluation classes. EuroSAT provides ten land-use and land-cover classes. Each task contains ten five-way episodes. Each selected class contributes ten support images and one query, so an episode contains 55 images and five queries.

The two tasks together use 1,100 distinct images, of which 100 are queries. Previously registered image sources are excluded before selection, and no source is reused across the fresh episodes. The C0 and C4 permission conditions use identical source and query images. C4 designates four of the ten support images per class as protected and leaves six mutable; it does not add extra support examples. Image anchors remain fixed class-text features in both conditions.

The complete source-selection record makes the finite evaluation population explicit. It includes dataset identity, class vocabulary, support/query role, and source index for each episode or slice. Shared query sources across encoders are intentional paired comparisons. The study does not interpret the resulting 960 text and 300 image query-model pairs as that many independent natural inputs, nor claim that these samples represent the entire public benchmark.

D.3 Protected information and task difficulty

Protected examples provide both a reference predictor and fixed information for some adapter interfaces. For generative text encoders they determine centering and class anchors. For native text and image encoders the primary class anchors come from fixed text, while protected examples are available to the protected-prototype reference. In all cases the protected reference and the adapter receive the same protected records for their respective condition.

The usefulness of these records is measured rather than assumed. On the fresh text panel, a protected prototype correctly classifies 682 of 960 query-model pairs; on image C4 it classifies 258 of 300. These figures exceed the clean accuracy of the primary distance adapter. They therefore inform classifier choice. The primary paired experiment instead evaluates the effect of normalization conditional on retaining that adapter's clean predictions.

E Comparison Scores and Secondary Variants

E.1 Distance references

Median distance removes the language-dependent term and uses $T(p_c)$ as its class score. It specializes FCert's configurable trimmed mean to the maximal trim $\tau(n_c)$ and recomputes that trim under our source-replacement permissions (Wang et al., 2024). The corresponding interval is $[T_p^-(r,s), T_p^+(r,s)]$. This defines the local comparison score, separately from the published fixed-class-count evaluation.

Mean distance uses the ordinary class mean of p_i. Its lower and upper bounds use the same deletion and insertion extremes, evaluated with the ordinary mean at the resulting class size. This rule can achieve high clean accuracy while having weak worst-case bounds under arbitrary replacement val-

ues. Its observed copy-attack harm is reported separately. A zero certified count is not evidence that every query can be attacked by the finite copy family.

Protected prototypes classify by distance to class means formed only from protected features. Class-text-only and native zero-shot scores compare the query with fixed class text. These predictors are constant under replacement of U, so a correct clean prediction remains correct for every attack in the stated threat model. Their certification follows from source independence, rather than the central-statistic interval calculation.

E.2 Protected-only calibration

The text comparison also includes a coefficient fitted only on protected sources. For each leave-one-out protected query, let b_{ic} be its distance to the protected class prototype formed without that query, and p_{ic} its median distance to the remaining protected examples of class c. Generative-model centering also excludes the held-out protected query for this fit. Define $\delta_{ic} = p_{ic} - b_{ic}$ and scores $s_{ic}(\alpha) = b_{ic} + \alpha\delta_{ic}$ for $\alpha \in [0, 1]$.

The coefficient minimizes the protected leave-one-out cross-entropy of logits $-s_{ic}(\alpha)$. Its derivative is

$$J'(\alpha) = \frac{1}{|C|} \sum_i \left[\delta_{i,y_i} - \sum_c \operatorname{softmax}(-s_i(\alpha))_c \delta_{ic} \right]. \tag{33}$$

The second derivative is the mean softmax-weighted variance of δ_i, hence nonnegative. If the derivative at zero is nonnegative, use zero; if the derivative at one is nonpositive, use one; otherwise solve its unique or interval-valued zero condition with a bracketed scalar root procedure. No test labels choose this coefficient. The resulting predictor remains independent of mutable support sources. Its fresh aggregate clean and certified counts both equal 682, although equal counts alone do not prove identity with the protected-prototype predictor.

E.3 Similarity scores and sign-aware bounds

The secondary similarity comparison changes the two-term score while preserving its own clean predictions under normalization. Using the same p, q, d, define the costs

$$R_c^{\mathrm{sim}} = L(n_c)\big(T(p_c) - 1\big) - \lambda(1 - d_c)\big(1 - T(q_c)\big), \tag{34}$$

$$\widetilde{R}_c^{\mathrm{sim}} = T(p_c) - 1 - \frac{\lambda}{L_*}(1 - d_c)\big(1 - T(q_c)\big). \tag{35}$$

Both are minimized over classes. The terms are negative similarities expressed through distance variables. With a common retained count, $R_c^{\mathrm{sim}} = L_* \widetilde{R}_c^{\mathrm{sim}}$, giving the same clean-preservation identity. This is a distinct score family from the primary distance rule.

Its class bounds account for the sign of the query-anchor similarity. Write the coefficient of $T(q_c)$ as $k_c = (\lambda/L_*)(1 - d_c)$ for the normalized score. A lower bound uses

$$T_p^-(r, s) - 1 - k_c + \min\{k_c T_q^-(r, s), k_c T_q^+(r, s)\}, \tag{36}$$

and the upper bound uses T_p^+ and the corresponding maximum. The original score multiplies the first term by $L(n_c')$ and replaces the coefficient by $\lambda(1 - d_c)$. The same shared-budget procedure then certifies the complete cost. Ignoring a negative k_c would reverse an inequality, which is why the interval implementation retains both endpoint products.

The fresh secondary results support the count mechanism without redefining the primary outcome. Text clean correctness is 655 for both similarity rules, certification increases from 501 to 520, and copy harm decreases from 48 to 17. Images preserve 226 correct predictions, increase certification from zero to 125, and reduce harm from 223 to 11. C0 and C4 give the same primary similarity counts on the reused image queries. All method-level rows appear below.

F Complete Fresh-Sample Results

The full tables retain all task, encoder, and permission combinations. Counts are computed on shared query sources within each panel. Every row reports clean correctness, certified correctness at budget

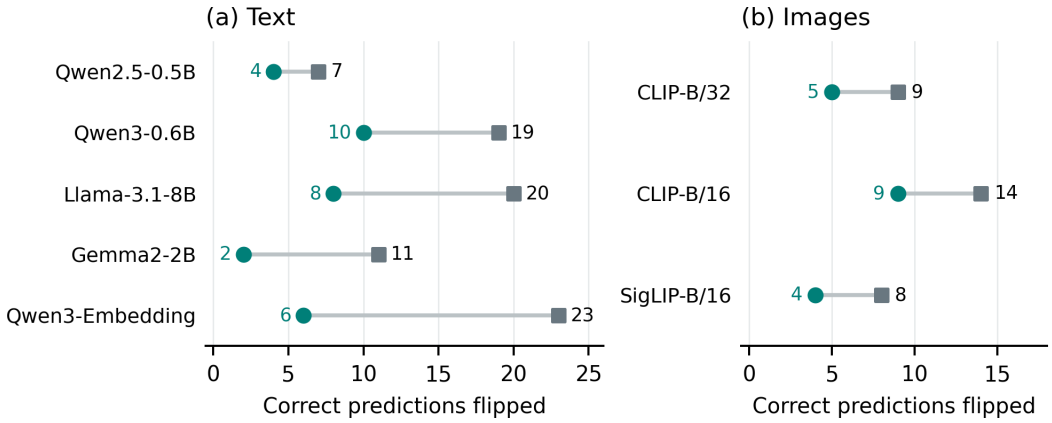


Figure 4: One-record copy attacks harm fewer correct predictions after normalization for every encoder in the distance study. Squares denote original scores and circles normalized scores. Lines connect paired counts on identical queries. Encoders within each panel share the same queries.

four, and harm from the one-record cross-class copy family. The original and normalized distance rules are the primary paired comparison; similarity rules and the other scores provide secondary and cross-method context. Within each task, the text rows aggregate the two support slices and the image rows aggregate ten episodes.

Table 7: Full text results on the fresh confirmation samples. Clean, certified-correct counts at budget four, and correct predictions flipped by one cross-class copy are reported for every scoring rule.

Method	N	Clean	Certified	Flipped
Gemma2-2B, AG News, C4				
Original distance	64	35	27	7
Normalized distance	64	35	28	1
Original similarity	64	47	43	1
Normalized similarity	64	47	44	0
Median distance	64	48	42	0
Mean distance	64	48	32	1
Protected prototype	64	48	48	0
Protected calibration	64	48	48	0
Gemma2-2B, Banking77 (four intents), C4				
Original distance	64	64	62	1
Normalized distance	64	64	63	0
Original similarity	64	64	64	0
Normalized similarity	64	64	64	0
Median distance	64	64	63	0
Mean distance	64	64	62	0
Protected prototype	64	64	64	0
Protected calibration	64	64	64	0

Continued on next page

Method	N	Clean	Certified	Flipped
Gemma2-2B, TweetEval emotion, C4				
Original distance	64	36	19	3
Normalized distance	64	36	20	1
Original similarity	64	47	33	1
Normalized similarity	64	47	33	0
Median distance	64	53	38	3
Mean distance	64	51	23	3
Protected prototype	64	46	46	0
Protected calibration	64	46	46	0
Llama-3.1-8B, AG News, C4				
Original distance	64	32	24	5
Normalized distance	64	32	28	2
Original similarity	64	42	36	3
Normalized similarity	64	42	36	4
Median distance	64	47	36	3
Mean distance	64	46	23	4
Protected prototype	64	44	44	0
Protected calibration	64	44	44	0
Llama-3.1-8B, Banking77 (four intents), C4				
Original distance	64	62	50	5
Normalized distance	64	62	54	2
Original similarity	64	61	61	0
Normalized similarity	64	61	61	0
Median distance	64	60	59	0
Mean distance	64	60	59	0
Protected prototype	64	61	61	0
Protected calibration	64	61	61	0
Llama-3.1-8B, TweetEval emotion, C4				
Original distance	64	40	19	10
Normalized distance	64	40	21	4
Original similarity	64	50	25	3
Normalized similarity	64	50	25	2
Median distance	64	51	25	4
Mean distance	64	55	5	6
Protected prototype	64	47	47	0
Protected calibration	64	47	47	0
Qwen2.5-0.5B, AG News, C4				
Original distance	64	26	17	4
Normalized distance	64	26	17	3
Original similarity	64	32	20	7
Normalized similarity	64	32	20	3
Median distance	64	39	8	3
Mean distance	64	36	0	8
Protected prototype	64	30	30	0
Protected calibration	64	30	30	0

Continued on next page

Method	N	Clean	Certified	Flipped
Qwen2.5-0.5B, Banking77 (four intents), C4				
Original distance	64	49	38	2
Normalized distance	64	49	42	1
Original similarity	64	56	54	1
Normalized similarity	64	56	54	0
Median distance	64	59	51	0
Mean distance	64	60	35	3
Protected prototype	64	57	57	0
Protected calibration	64	57	57	0
Qwen2.5-0.5B, TweetEval emotion, C4				
Original distance	64	23	11	1
Normalized distance	64	23	11	0
Original similarity	64	25	7	2
Normalized similarity	64	25	8	0
Median distance	64	38	3	9
Mean distance	64	30	0	12
Protected prototype	64	26	26	0
Protected calibration	64	26	26	0
Qwen3-0.6B, AG News, C4				
Original distance	64	28	16	7
Normalized distance	64	28	16	2
Original similarity	64	30	22	2
Normalized similarity	64	30	22	2
Median distance	64	30	9	4
Mean distance	64	27	0	3
Protected prototype	64	32	32	0
Protected calibration	64	32	32	0
Qwen3-0.6B, Banking77 (four intents), C4				
Original distance	64	36	23	2
Normalized distance	64	36	23	2
Original similarity	64	38	26	0
Normalized similarity	64	38	26	0
Median distance	64	40	21	3
Mean distance	64	41	9	3
Protected prototype	64	40	40	0
Protected calibration	64	40	40	0
Qwen3-0.6B, TweetEval emotion, C4				
Original distance	64	29	8	10
Normalized distance	64	29	10	6
Original similarity	64	27	8	6
Normalized similarity	64	27	8	4
Median distance	64	38	5	4
Mean distance	64	40	0	11
Protected prototype	64	30	30	0
Protected calibration	64	30	30	0

Continued on next page

Method	N	Clean	Certified	Flipped
Qwen3-Embedding, AG News, C4				
Original distance	64	42	35	4
Normalized distance	64	42	38	1
Original similarity	64	43	40	1
Normalized similarity	64	43	42	0
Median distance	64	50	42	0
Mean distance	64	52	0	3
Protected prototype	64	53	53	0
Protected calibration	64	53	53	0
Class-text only	64	51	51	0
Qwen3-Embedding, Banking77 (four intents), C4				
Original distance	64	54	38	8
Normalized distance	64	54	45	4
Original similarity	64	54	37	11
Normalized similarity	64	54	48	0
Median distance	64	64	61	0
Mean distance	64	64	0	1
Protected prototype	64	62	62	0
Protected calibration	64	62	62	0
Class-text only	64	58	58	0
Qwen3-Embedding, TweetEval emotion, C4				
Original distance	64	39	21	11
Normalized distance	64	39	28	1
Original similarity	64	39	25	10
Normalized similarity	64	39	29	2
Median distance	64	52	34	2
Mean distance	64	51	0	4
Protected prototype	64	42	42	0
Protected calibration	64	42	42	0
Class-text only	64	41	41	0

Table 8: Full image results on the fresh confirmation samples. Clean, certified-correct counts at budget four, and correct predictions flipped by one cross-class copy are reported for every scoring rule.

Method	N	Clean	Certified	Flipped
CLIP-B/16, CIFAR-FS, C0				
Original distance	50	45	28	5
Normalized distance	50	45	28	4
Original similarity	50	45	0	45
Normalized similarity	50	45	30	1
Median distance	50	43	27	3
Mean distance	50	45	0	3
Zero-shot	50	47	47	0

Continued on next page

Method	N	Clean	Certified	Flipped
CLIP-B/16, CIFAR-FS, C4				
Original distance	50	45	28	5
Normalized distance	50	45	28	4
Original similarity	50	45	0	45
Normalized similarity	50	45	30	1
Median distance	50	43	27	3
Mean distance	50	45	0	3
Zero-shot	50	47	47	0
Protected prototype	50	46	46	0
CLIP-B/16, EuroSAT, C0				
Original distance	50	24	8	9
Normalized distance	50	24	8	5
Original similarity	50	27	0	27
Normalized similarity	50	27	8	1
Median distance	50	35	8	4
Mean distance	50	37	0	6
Zero-shot	50	27	27	0
CLIP-B/16, EuroSAT, C4				
Original distance	50	24	8	9
Normalized distance	50	24	8	5
Original similarity	50	27	0	27
Normalized similarity	50	27	8	1
Median distance	50	35	8	4
Mean distance	50	37	0	6
Zero-shot	50	27	27	0
Protected prototype	50	35	35	0
CLIP-B/32, CIFAR-FS, C0				
Original distance	50	46	31	3
Normalized distance	50	46	32	2
Original similarity	50	47	0	45
Normalized similarity	50	47	34	1
Median distance	50	44	25	2
Mean distance	50	43	0	1
Zero-shot	50	49	49	0
CLIP-B/32, CIFAR-FS, C4				
Original distance	50	46	31	3
Normalized distance	50	46	32	2
Original similarity	50	47	0	45
Normalized similarity	50	47	34	1
Median distance	50	44	25	2
Mean distance	50	43	0	0
Zero-shot	50	49	49	0
Protected prototype	50	47	47	0
CLIP-B/32, EuroSAT, C0				
Original distance	50	20	8	6
Normalized distance	50	20	8	3
Original similarity	50	26	0	25
Normalized similarity	50	26	8	4
Median distance	50	34	7	0
Mean distance	50	33	0	4
Zero-shot	50	27	27	0

Continued on next page

Method	N	Clean	Certified	Flipped
CLIP-B/32, EuroSAT, C4				
Original distance	50	20	8	6
Normalized distance	50	20	8	3
Original similarity	50	26	0	25
Normalized similarity	50	26	8	4
Median distance	50	34	7	0
Mean distance	50	33	0	3
Zero-shot	50	27	27	0
Protected prototype	50	38	38	0
SigLIP-B/16, CIFAR-FS, C0				
Original distance	50	47	39	2
Normalized distance	50	47	40	0
Original similarity	50	47	0	47
Normalized similarity	50	47	39	0
Median distance	50	45	31	1
Mean distance	50	46	0	3
Zero-shot	50	48	48	0
SigLIP-B/16, CIFAR-FS, C4				
Original distance	50	47	39	2
Normalized distance	50	47	40	0
Original similarity	50	47	0	47
Normalized similarity	50	47	39	0
Median distance	50	45	31	1
Mean distance	50	46	0	3
Zero-shot	50	48	48	0
Protected prototype	50	48	48	0
SigLIP-B/16, EuroSAT, C0				
Original distance	50	21	8	6
Normalized distance	50	21	8	4
Original similarity	50	34	0	34
Normalized similarity	50	34	6	4
Median distance	50	38	10	3
Mean distance	50	40	0	7
Zero-shot	50	28	28	0
SigLIP-B/16, EuroSAT, C4				
Original distance	50	21	8	6
Normalized distance	50	21	8	4
Original similarity	50	34	0	34
Normalized similarity	50	34	6	4
Median distance	50	38	10	3
Mean distance	50	40	0	7
Zero-shot	50	28	28	0
Protected prototype	50	44	44	0

F.1 CROSS-METHOD CONTEXT

Reference comparisons place the paired repair in the context of classifier choice. Table 9 uses the same features as the primary distance rules. Median distance has more text certificates, while normalized distance has more image certificates. Protected references remain stronger in both distance panels; zero-shot is stronger on images. These comparisons retain the distinction between choosing a new classifier and preserving the original clean decisions.

Table 9: Cross-method context at replacement budget four. Primary image distance counts are identical in C0 and C4. Protected prototypes require C4; zero-shot uses fixed class text. Immutable references have identical clean-correct and certified-correct counts.

	Text ($N = 960$)		Images ($N = 300$)	
Method	Clean	Certified	Clean	Certified
Original distance	595	408	203	122
Normalized distance	595	444	203	124
Median distance	733	497	239	108
Mean distance	725	248	244	0
Protected prototype	682	682	258	258
Zero-shot	n/a	n/a	226	226

F.2 ALL REPLACEMENT BUDGETS

The budget table reports the complete primary certificate curve for each encoder. The fixed-count row applies the same interval construction restricted to profiles that preserve class sizes. Its score-scale identity holds for both the original and normalized predictors. The variable-count rows use the larger replacement threat. Image C4 has the same primary curves as C0 for these sampled episodes, so the table presents C0 once rather than duplicating identical numbers.

Table 10: Certified-correct counts across all budgets for the primary scoring rule. Fixed-count values are identical for the original and normalized rules.

Encoder	Rule	$m = 0$	1	2	3	4
Qwen2.5-0.5B	Original	98	91	85	75	66
	Normalized	98	93	88	82	70
	Fixed counts	98	94	88	82	70
Qwen3-0.6B	Original	93	74	65	53	47
	Normalized	93	82	69	54	49
	Fixed counts	93	83	71	54	49
Llama-3.1-8B	Original	134	113	105	99	93
	Normalized	134	120	113	110	103
	Fixed counts	134	120	114	110	103
Gemma2-2B	Original	135	124	122	115	108
	Normalized	135	133	130	121	111
	Fixed counts	135	133	130	123	111
Qwen3-Embedding	Original	135	112	105	100	94
	Normalized	135	126	123	120	111
	Fixed counts	135	127	124	120	111
CLIP-B/32	Original	66	57	53	44	39
	Normalized	66	60	54	47	40
	Fixed counts	66	60	54	47	41

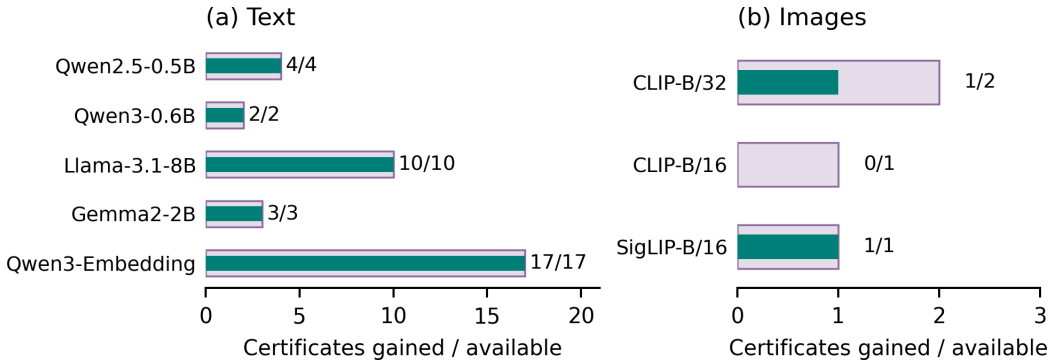


Figure 5: The fixed-count gap bounds the distance-score increase available within this certificate family. Filled bars show the observed increase in certified-correct predictions at budget four; outlined bars show H_4. Labels report gained/available counts. The gap does not measure real attack risk or the capacity of other scoring rules.

Table 10 (continued)

Encoder	Rule	$m = 0$	1	2	3	4
CLIP-B/16	Original	69	55	52	45	36
	Normalized	69	59	54	48	36
	Fixed counts	69	59	54	48	37
SigLIP-B/16	Original	68	60	58	52	47
	Normalized	68	62	59	53	48
	Fixed counts	68	62	59	54	48

F.3 The two unrecovered fixed-count certificates

The text panel closes its fixed-count certificate gap at budget four, while the image panel retains two residual cases. After the primary analysis, we select every correctly classified query in $\mathcal{C}_{\text{fix}} \setminus \mathcal{C}_{\text{norm}}$ at that budget. Both cases evaluate one shared CIFAR-FS query with two CLIP encoders in C0. Thus the remaining gap concerns one distinct input; no new method or sample is selected.

Table 11 records a minimizing profile for each case. A profile lists (r_w, s_w, r_c, s_c), and every margin uses the normalized distance score. The fixed-count minimizers are $(3, 3, 1, 1)$ for CLIP-B/16 and $(0, 0, 4, 4)$ for CLIP-B/32. Both variable-count representatives use four removals and four insertions within the winner-rival pair, so their budgets are globally feasible without a third class. The ordered class vocabulary is saved with each case in `analysis/binding_and_timing.json`.

Table 11: Every unrecovered fixed-count certificate at budget four. Rows refer to one CIFAR-100 training image in the CIFAR-FS evaluation classes. Winner and rival indices (w, c) are zero-based in the corresponding class vocabulary. Both rows share one query.

Encoder	Row	(w, c)	Fixed	Variable	Profile
CLIP-B/16	47390	$(2, 0)$	0.025946	-0.006101	$(3, 4, 1, 0)$
CLIP-B/32	47390	$(2, 3)$	0.024394	-0.077516	$(0, 1, 4, 3)$

The residual comes from changes in the retained statistics after the explicit multiplier is fixed. Relative to the fixed-count minimizer, the winner's upper bound increases by 0.10278 and the rival's lower bound by 0.07073 for CLIP-B/16. The corresponding changes are 0.11384 and 0.01193 for CLIP-B/32. These differences explain the lost positive margins. Feasible counts alone do not make the independently chosen scalar extrema jointly attainable, so these profiles are certificate diagnostics, not new executable attacks.

G Paired Uncertainty and Source-Cluster Results

G.1 Estimands and repeated inputs

The paired effects compare methods on the same query-model pairs. For cluster b and model j, let G_{bj} be the difference in certified-correct counts and A_{bj} the decrease in correctly classified queries flipped by the copy family. The reported panel effects divide $\sum_{bj} G_{bj}$ and $\sum_{bj} A_{bj}$ by the total number of query-model pairs. This defines a finite-panel summary, conditional on the selected encoders and tasks.

Clusters carry the shared randomness of the support set. Text has two slices in each of three tasks, or six source clusters. Images have ten episodes in each of two tasks, or twenty clusters. The same query is evaluated by several encoders, and every query in a cluster uses the same support set. C0 and C4 also reuse image sources. Treating these observations as independent would understate uncertainty and change the inferential unit.

G.2 Task-stratified paired bootstrap

The bootstrap samples source clusters with replacement within each task, retaining the original number of clusters in that task. For a sampled cluster, the same multiplicity applies to all encoders and to both paired methods. The procedure sums the resulting paired effects and divides by the fixed panel size. We use 20,000 replicates and report the 2.5th and 97.5th empirical percentiles.

The fresh text certified-accuracy gain is $36/960 = 3.75$ percentage points with interval $[3.5417, 3.9583]$. The harm reduction is $50/960 = 5.2083$ points with interval $[4.1667, 6.2500]$. Images have a certified gain of $2/300 = 0.6667$ points with interval $[0, 1.6667]$ and harm reduction $13/300 = 4.3333$ points with interval $[1.6667, 7.3333]$. The image C4 values are identical for the primary score.

The intervals describe resampling within this finite task design. In particular, two text slices per task provide limited support for estimating an underlying distribution of tasks or future support sets. The original analysis criterion required positive directions in both metrics for text and image C0, with a stronger outcome only when all four interval lower bounds were strictly positive. The direction criterion is met; the stronger criterion is not, because image certification includes zero. The main text reports this distinction without changing the registered budget or selecting a different primary score.

G.3 Complete cluster summaries

The following table exposes the paired results at the resampling unit. Text counts aggregate five encoders per slice, and image counts aggregate three encoders per episode. The fixed-count column supplies the same-family reference used to compute H_4. Displaying the shared source clusters makes both effect heterogeneity and the effective number of sampled support sets visible.

Table 12: Paired source-cluster results, aggregating encoders within each shared slice or episode. Harm counts initially correct predictions flipped by a copy. Image conditions reuse queries and are listed separately.

Cluster	N	Clean	Cert. old/new	Fixed	Harm old/new
CIFAR-FS, C0					
0	15	14	10/10	12	1/0
1	15	13	7/8	8	1/1
2	15	14	12/12	12	1/1
3	15	13	8/8	8	2/1
4	15	15	11/11	11	1/0
5	15	15	7/7	7	2/1
6	15	10	9/9	9	1/1
7	15	15	13/14	14	0/0
8	15	14	13/13	13	0/0
9	15	15	8/8	8	1/1
EuroSAT, C0					
0	15	10	3/3	3	4/0
1	15	9	3/3	3	3/1
2	15	6	2/2	2	1/0
3	15	7	0/0	0	2/2
4	15	6	3/3	3	3/3
5	15	4	3/3	3	1/1
6	15	7	2/2	2	2/2
7	15	8	2/2	2	4/2
8	15	4	3/3	3	0/0
9	15	4	3/3	3	1/1
CIFAR-FS, C4					
0	15	14	10/10	12	1/0
1	15	13	7/8	8	1/1
2	15	14	12/12	12	1/1
3	15	13	8/8	8	2/1
4	15	15	11/11	11	1/0
5	15	15	7/7	7	2/1
6	15	10	9/9	9	1/1
7	15	15	13/14	14	0/0
8	15	14	13/13	13	0/0
9	15	15	8/8	8	1/1
EuroSAT, C4					
0	15	10	3/3	3	4/0
1	15	9	3/3	3	3/1
2	15	6	2/2	2	1/0
3	15	7	0/0	0	2/2
4	15	6	3/3	3	3/3
5	15	4	3/3	3	1/1
6	15	7	2/2	2	2/2
7	15	8	2/2	2	4/2
8	15	4	3/3	3	0/0
9	15	4	3/3	3	1/1
AG News, C4					
0	160	81	60/65	65	15/4
1	160	82	59/62	62	12/5
Banking77 (four intents), C4					
0	160	126	104/112	112	4/2
1	160	139	107/115	115	14/7
TweetEval emotion, C4					
0	160	89	47/53	53	18/9
1	160	78	31/37	37	17/3

H COUNT-NORMALIZED EXPONENTIAL CACHE ADAPTATION

H.1 CACHE CONSTRUCTION AND FIXED COEFFICIENTS

The cache study evaluates the training-free score and cache construction of Tip-Adapter (Zhang et al., 2022a).[1] Each support image produces ten randomly augmented views. We average their raw projected CLIP image features and then normalize the average to obtain one cache key. Each support source therefore contributes one key and one label, rather than ten separately replaceable records. Query images use deterministic preprocessing. Class anchors are normalized CLIP text features and remain fixed throughout replacement.

Support augmentation uses a random resized crop to 224×224, with area scale in $[0.5, 1]$, aspect ratio in $[3/4, 4/3]$, bicubic interpolation, and horizontal flipping with probability 0.5. Each source-view pair has a fixed random seed, so batch boundaries and resumed execution do not change the augmentation. Query preprocessing uses the model's deterministic resize and center crop. Both paths use the CLIP image mean and standard deviation. The two cache rules share every projected feature and preprocessing choice.

We fix the exponential coefficient $\beta = 1$ and cache weight $\alpha = 2$ for both encoders and both tasks. These are the initial values in the public Tip-Adapter EuroSAT configuration. We evaluate this fixed operating point without its validation-set parameter search. The study uses the full training-free cache computation, not the separately fine-tuned Tip-Adapter-F variant. Its five-way episodes and CLIP vision-transformer backbones define the comparison here; published optimized benchmark accuracies use different selection protocols and are not comparison rows.

The original cache score can be written as a fixed text score plus class mass times average support evidence:

$$k_i = \exp[-\beta(1 - u^\top z_i)], \quad \bar{k}_c = \frac{1}{n_c} \sum_{i:y_i=c} k_i, \quad S_c = 100 u^\top t_c + \alpha n_c \bar{k}_c. \tag{37}$$

Normalization replaces n_c by its initial value $n_{0,c}$ in this last expression. It neither changes α or β nor modifies the fixed text term. Every initial class has ten sources, and the prediction selects the largest score, breaking ties by class index.

H.2 PRESERVATION AND A COUNT-ONLY CONSTRUCTION

Cache-score preservation follows by direct substitution. For any positive initial counts, including unequal ones, $n_c = n_{0,c}$ gives

$$\widetilde{S}_c = 100 u^\top t_c + \alpha \frac{n_{0,c}}{n_c} \sum_{i:y_i=c} k_i = S_c. \tag{38}$$

The entire class-score vector is identical, so every clean prediction and tie is preserved. The identity also holds after every count-preserving replacement, even when support features change. Thus fixed-count attacks provide a control under which the two cache rules cannot differ.

A constant-kernel construction isolates the extra class-mass effect. Consider two classes whose support keys all give kernel value $k_0 > 0$ for the query, with equal initial class counts. Let the fixed text-score advantage of the winner over the rival be $\delta > 0$. Moving one record from the winner to the rival leaves both kernel means unchanged. The original margin becomes $\delta - 2\alpha k_0$, while the normalized margin remains δ. If $0 < \delta < 2\alpha k_0$, only the original prediction changes.

This construction is attainable in the bounded feature interface. Take $u = e_1$ and support keys orthogonal to u, so $k_0 = e^{-\beta}$. Fixed unit text anchors can have first-coordinate difference $\delta/100$. Other class scores can be placed below the two competing scores. The construction establishes a score-level mechanism, independently of whether a particular pretrained encoder maps natural images to these exact vectors. The benchmark-copy experiments use actual cached features from public images.

[1] The official implementation supplies the cache construction, EuroSAT configuration, and class names and prompts.

H.3 SOURCE-LEVEL SCORE BOUNDS

The exponential kernel has a fixed range under the normalized feature interface. Since $-1 \le u^\top z_i \le 1$,

$$a := e^{-2\beta} \le k_i \le 1. \tag{39}$$

For a given class, let C_c denote its protected positions and let the M_c mutable kernel values in increasing order be $b_1 \le \cdots \le b_{M_c}$. If r records are removed and s inserted, the resulting class size is $n'_c = |C_c| + M_c - r + s$. The kernel sum is bounded by

$$K_c^-(r,s) = \sum_{i \in C_c} k_i + \sum_{j=1}^{M_c - r} b_j + sa, \tag{40}$$

$$K_c^+(r,s) = \sum_{i \in C_c} k_i + \sum_{j=r+1}^{M_c} b_j + s. \tag{41}$$

The lower sum removes the largest mutable values and inserts the minimum kernel value. The upper sum removes the smallest and inserts the maximum. Every legal remaining subset and inserted value is enclosed, proving both inequalities.

The bounds propagate through the complete score because both cache coefficients are positive. The normalized score lies in

$$\left[100u^\top t_c + \alpha \frac{n_{0,c}}{n'_c} K_c^-(r,s), \quad 100u^\top t_c + \alpha \frac{n_{0,c}}{n'_c} K_c^+(r,s)\right]. \tag{42}$$

Dropping $n_{0,c}/n'_c$ gives the original-score interval. For each clean winner and rival, we minimize the winner's lower score minus the rival's upper score over the same shared removal and insertion profiles as Equation 1. A strictly positive minimum for every rival certifies the prediction. Five classes with ten sources each, or six mutable sources under C4, satisfy the pair-completion conditions at all budgets up to four.

These bounds certify individual queries under arbitrary bounded-feature replacements. Their kernel endpoints are attained by $z = -u$ and $z = u$ in a free unit-feature domain. We do not need to assume that a natural input realizes either endpoint. Different queries may also require different extremal keys. Consequently, per-query bounds should not be interpreted as one jointly realizable worst-case cache for all queries. The shared attacks below evaluate complete, common cache states directly.

H.4 INDEPENDENT SOURCE SELECTION AND SCOPE

The new sample panel contains 40 source-disjoint support sets. For each of CIFAR-FS and EuroSAT, we select 20 five-way episodes, with ten support images and two queries per class. There are 2,000 distinct support images and 400 distinct query images. CLIP-B/32 and CLIP-B/16 share all sources, giving 800 query-model pairs. C0 permits replacement of every support source. C4 protects the first four sources per class on the same images and does not add independent samples.

CIFAR-FS uses the same 20 held-out classes as the distance study, with their names listed in `fewshot_results.zip/protocol.json` and disjoint support and query sources selected from those classes (Bertinetto et al., 2019). For EuroSAT (Helber et al., 2019), support images and queries use the training and test splits distributed by `timm/eurosat-rgb`, respectively. These splits follow the definitions in Google's remote-sensing repository.[2] The selection excludes all image sources used in the registered earlier studies and the primary distance confirmation. Source identities, roles, dataset revisions, and model revisions are included in the selection record. The cache configuration was developed on earlier disjoint features and fixed before extracting these new readouts.

The class-text prompts follow the dataset semantics. CIFAR-FS uses `a photo of a {class}.`, replacing underscores in class names with spaces. EuroSAT uses `a centered satellite photo of {class}.` with the public Tip-Adapter names: Annual Crop Land,

[2] `https://huggingface.co/datasets/timm/eurosat-rgb`

Forest, Herbaceous Vegetation Land, Highway or Road, Industrial Buildings, Pasture Land, Permanent Crop Land, Residential Buildings, River, and Sea or Lake. The episode vocabulary fixes the order of class scores and tie handling.

The attacked interface is the frozen cache, including each key and its label. Copying a source means copying this complete encoded record into a mutable position. It does not mean independently rerunning stochastic augmentation for each duplicate. The certificate also covers any new image whose fixed preprocessing produces a bounded key, but the executable attacks use the narrower family of stored natural-source copies. Neither the encoder nor the query features are optimized by the attacker.

H.5 Individual-copy and shared-cache attacks

The one-record attack exhaustively enumerates the cross-class copy family separately for every query and classifier. A candidate replaces one mutable position with an original support key and that source's correct label. C0 has $5 \cdot 4 \cdot 10 \cdot 10 = 2{,}000$ candidates per query; C4 has $1{,}200$. A query contributes to copy harm if it was initially correct and at least one candidate changes its prediction. The search is complete within this one-record copy family, not within arbitrary feature replacement.

The shared attack uses one common changed cache for every query in an episode. We split the queries before attack selection: the first query of each class is available to the attack, and the second is held out. The attack chooses up to four different mutable positions to replace. At each step it enumerates cross-class copies from the original support pool, selects the candidate that flips the most clean predictions on the five selection queries, and breaks ties by the smallest mean clean-winner margin. Remaining ties use source order. It stops if no candidate improves this lexicographic objective. Query labels do not participate in selection.

We run this procedure against the original and normalized classifier, then evaluate each resulting complete cache on both classifiers. This gives a two-by-two harm matrix for selection plus held-out queries, and a separate matrix for held-out queries alone. The same copy path is replayed on both scoring rules. Previously changed positions cannot be selected again, protected positions remain unchanged, and the donor carries its original label. The attack is greedy and has no random restarts, so its observed harm is a lower bound on the optimum over legal shared copies.

The primary transfer statistic takes the worse of the two terminal attacks separately for each episode, encoder, and evaluated classifier. If $h_{e,b,a,k}$ is the number of initially correct held-out queries harmed in episode e for encoder b, attack target a, and evaluated classifier k, its total is

$$H_k = \sum_{e,b} \max_{a \in \{\text{original,normalized}\}} h_{e,b,a,k}. \tag{43}$$

Each term corresponds to one legal shared cache for that encoder. Taking a maximum independently for each query would combine incompatible caches. The maximum uses held-out labels only to evaluate the two already selected attacks, not to construct either path. It is a finite-attack-bank oracle statistic; the poisoned caches may differ across encoders. We additionally report the complete cross-evaluation matrices so both attack choices remain visible.

Saved intermediate states provide a further check at budgets up to four. Taking the most harmful complete state over both paths and all their prefixes gives 208 versus 9 harmed held-out pairs in C0, compared with the terminal totals of 206 versus 9. The corresponding all-query totals are 456 versus 27 for both prefix and terminal states. In C4, held-out prefix totals are 205 versus 7, compared with 204 versus 7 at the terminal states. Every prefix is a legal shared cache. In C0, including prefixes increases the paired held-out harm reduction by one query-model pair in each of two EuroSAT episodes, from 49.25 to 49.75 percentage points overall. The primary statistic and its bootstrap intervals use terminal states; the prefix results are reported separately.

H.6 Complete paired results

All four encoder-task combinations improve certification and copy resistance in the C0 primary condition. Table 13 reports each combination and the repeated C4 condition. The pooled C0 clean count is 700, compared with 643 for zero-shot and 695 for a full-support prototype. On CIFAR-FS, the cache has 393 correct predictions compared with 391 for zero-shot; on EuroSAT, the corresponding

Table 13: Complete cache-study results. Each row has 200 query-model pairs for clean accuracy, budget-four certification, and one-copy harm, and 100 held-out pairs for shared harm. Slashes give original/normalized counts. Clean counts are identical. Shared harm takes the worse terminal attack per episode for that encoder; zero-shot is independent of all editable sources.

Task	Encoder	Clean	Zero-shot	Cert.	Copy harm	Shared harm
All sources mutable						
CIFAR-FS	CLIP-B/32	196	193	0/104	32/1	44/0
CIFAR-FS	CLIP-B/16	197	198	0/103	43/0	47/0
EuroSAT	CLIP-B/32	143	122	0/10	117/13	53/4
EuroSAT	CLIP-B/16	164	130	0/9	133/3	62/5
Four protected sources per class						
CIFAR-FS	CLIP-B/32	196	193	0/105	32/1	43/0
CIFAR-FS	CLIP-B/16	197	198	0/104	43/0	47/0
EuroSAT	CLIP-B/32	143	122	0/10	117/12	51/3
EuroSAT	CLIP-B/16	164	130	0/9	133/3	63/4

Table 14: Cache certified-correct counts at every replacement budget. Each task-condition group contains 400 query-model pairs. C0 and C4 reuse the same images and initial predictions.

Condition	Task / score	$m = 0$	$m = 1$	$m = 2$	$m = 3$	$m = 4$
C0	CIFAR-FS, original	393	304	132	7	0
C0	CIFAR-FS, normalized	393	380	335	280	207
C0	EuroSAT, original	307	53	4	0	0
C0	EuroSAT, normalized	307	156	65	31	19
C4	CIFAR-FS, original	393	304	136	7	0
C4	CIFAR-FS, normalized	393	380	336	282	209
C4	EuroSAT, original	307	53	4	0	0
C4	EuroSAT, normalized	307	158	65	34	19

counts are 307 and 252. These are fixed-parameter, five-way results. Normalization preserves every clean prediction, not merely these totals.

The original cache has no certified-correct queries at budget four, whereas normalization certifies 226 in C0 and 228 in C4. The gain is larger on CIFAR-FS than EuroSAT: 207 versus 19 in C0. Table 14 retains all budgets. Single-copy harm falls from 325 to 17 in C0 and from 325 to 16 in C4. These outcomes describe separate aspects of robustness: certificates cover the larger arbitrary-key threat, and successful copies establish susceptibility to actual stored sources.

Shared-copy resistance improves against attacks selected on either classifier. In C0, the all-query harm matrix is $\left(\begin{smallmatrix} 453 & 8 \\ 206 & 27 \end{smallmatrix}\right)$ and the held-out matrix is $\left(\begin{smallmatrix} 201 & 2 \\ 102 & 9 \end{smallmatrix}\right)$, with rows denoting attack targets and columns evaluated classifiers. The corresponding terminal worst-of-two held-out totals are 206 and 9, rather than the maxima of the pooled matrix columns. Table 15 reports both tasks and permission conditions, including the attack targeted at normalization.

Protected and zero-shot classifiers provide a separate comparison with immutable predictions. In C4, protected prototypes classify 655 of 800 queries correctly: 378 on CIFAR-FS and 277 on EuroSAT. Their certified-correct counts equal these clean counts because the attacker cannot change their inputs. Zero-shot has 643 correct and certified predictions in both conditions. Thus cache normalization preserves a higher clean accuracy while improving the robustness of that adapted predictor; it does not exceed these immutable references in budget-four certified accuracy.

A post hoc comparison identifies which adapted decisions add value over zero-shot. The cache corrects 67 query-model pairs that zero-shot gets wrong, while losing ten of its correct predictions. The net gain is therefore 57. All 67 additions are vulnerable to an original one-record copy attack; only 11 remain vulnerable after normalization. CIFAR-FS contributes three additions and one loss, with copy harm reduced from three to zero. EuroSAT contributes 64 additions and nine losses, with harm reduced from 64 to 11. Thus 56 additions survive every single copy: three on CIFAR-FS and

Table 15: Terminal shared-cache correct-prediction harm under both attack targets. All-query columns use 400 pairs per task and condition; held-out columns use 200. Each row evaluates the same complete attacked caches on both classifiers. C0 and C4 are repeated permission conditions.

Condition	Task	Attack target	All queries		Held-out queries	
			Original	Norm.	Original	Norm.
C0	CIFAR-FS	Original	206	1	89	0
C0	CIFAR-FS	Normalized	66	2	31	0
C0	EuroSAT	Original	247	7	112	2
C0	EuroSAT	Normalized	140	25	71	9
C4	CIFAR-FS	Original	209	1	90	0
C4	CIFAR-FS	Normalized	43	1	19	0
C4	EuroSAT	Original	249	9	113	2
C4	EuroSAT	Normalized	134	19	66	6

Table 16: Paired cache-study gains in percentage points. Overall intervals target simultaneous 95% coverage across the three prespecified metrics; task-level intervals are individual descriptive 95% intervals. Positive harm reduction favors normalization.

Scope	Certified accuracy	One-copy harm	Held-out shared harm
Overall	$28.25\ [24.00, 32.50]$	$38.50\ [34.38, 42.63]$	$49.25\ [44.00, 54.75]$
CIFAR-FS	$51.75\ [45.25, 58.50]$	$18.50\ [14.25, 23.00]$	$45.50\ [39.50, 52.00]$
EuroSAT	$4.75\ [2.50, 7.25]$	$58.50\ [53.50, 63.50]$	$53.00\ [47.00, 59.00]$

53 on EuroSAT. Of the 67 additions, 12 receive a normalized certificate at budget one, comprising one on CIFAR-FS and eleven on EuroSAT, and none at budgets two through four. This separates the observed copy resistance of useful adapted decisions from their remaining certification gap.

H.7 Paired uncertainty and decision criteria

Uncertainty is computed over support episodes with encoder pairing preserved. Each bootstrap replicate samples 20 episodes with replacement within each task and includes both encoders for every selected episode. We use 20,000 replicates. The three prespecified primary effects are the increase in certified accuracy, the decrease in one-copy harm, and the decrease in held-out shared harm. The first two use 800 query-model pairs; the third uses 400. These denominators count paired evaluations, not independent images.

The simultaneous intervals allocate the family error rate across the three effects. We use percentile endpoints $0.05/(2 \cdot 3)$ and $1 - 0.05/(2 \cdot 3)$ for each effect, corresponding to individual 98.33% intervals and a Bonferroni simultaneous 95% target. As with ordinary cluster bootstrap, finite-sample coverage is approximate. Table 16 reports the primary intervals and individual descriptive 95% intervals by task. No additional samples or parameters were selected after observing these outcomes.

The observed outcomes satisfy the criteria fixed before feature extraction. Every clean prediction is preserved, all three simultaneous lower bounds are positive, neither task reverses an effect direction, and at least one task has a clean gain over zero-shot. The corresponding direction is positive on both tasks. These results concern the new cache study; they do not change the distance study's original image interval or its stronger criterion reported in Section 5.

H.8 Computation and saved-feature reproduction

The workload uses frozen forward passes and reusable cache statistics. Two encoders process ten views of 2,000 support images and one view of 400 queries, giving 40,800 image forward evaluations, plus 60 class-text evaluations. No encoder training or cache fine-tuning is performed. The normalization uses one positive classwise factor after accumulating kernel sums and shares the same encodings and score terms as the original classifier.

Saved float32 image-view and class-text readouts, source maps, and per-query records support reconstruction without further model inference. Source records retain the view indices and support/query roles. Cache reconstruction averages the support-view readouts and normalizes the keys in float64. It then calculates class scores and intervals, enumerates all one-record copies, and replays both shared attack selectors, including the held-out split. Result summaries are generated from these records. Numerical comparisons use float64; the implementation does not perform directed rounding, so numerical certificates are not formal floating-point proofs.

I General Coefficient-Count Sensitivity

The finite-count difference in Equation 5 and the reference correction in Equation 6 separate explicit coefficients from support-dependent statistics. We prove their gap and preservation properties, then account for the statistic changes that remain under real source replacements.

I.1 A fixed component representation

Fix the query, encoder, preprocessing, class anchors, vocabulary, and class-index tie rule. Let D_0 be the reference support, with initial counts $n_{0,c} \geq 1$ and mutable positions U_c. Consider a lower-is-better score

$$R_c(D, x) = \mathbf{w}_c(n_c)^\top \mathbf{z}_c(D, x), \qquad \mathbf{w}_c(n_c), \mathbf{z}_c(D, x) \in \mathbb{R}^{J_c}. \tag{44}$$

Here $\mathbf{z}_c(D, x)$ is the vector of class-level component statistics, not a source feature z_i. Its coordinates can include means, trimmed means, or fixed terms. A fixed bias is represented by a constant coordinate. The coefficient function $\mathbf{w}_c$ is specified by the scoring rule and depends on the current class count. The representation is fixed before the comparison. Absorbing a count coefficient into a component would define a different normalization.

Writing a score in this form does not make its statistics count-invariant. Copying can change an empirical distribution, and changing a class size can change trimming ranks. We first hold $\mathbf{z}_{0,c} := \mathbf{z}_c(D_0, x)$ fixed to isolate the explicit coefficient dependence. We then account for both changes in an actual replacement.

I.2 Legal integer changes and exact gap conditions

The possible net count changes under the source-replacement budget are

$$\mathcal{N}_m = \left\{ \delta \in \mathbb{Z}^K : \begin{array}{c} \sum_c \delta_c = 0, \quad \sum_c (\delta_c)_+ \leq m, \\ n_{0,c} + \delta_c \geq 1, \quad -\delta_c \leq |U_c| \quad \text{for every } c \end{array} \right\}, \tag{45}$$

where $(a)_+ = \max(a, 0)$. These conditions are necessary because $\delta_c = s_c - r_c$, total removals equal total insertions, and $r_c \leq |U_c|$. They are sufficient for the counts: remove $(-\delta_c)_+$ distinct mutable records from each losing class and match them to the $(\delta_c)_+$ insertions in gaining classes. The two totals agree and do not exceed m. This constructs a legal replacement profile with the stated nonempty final classes. It need not preserve the component statistics, so a fixed-statistic count comparison need not correspond to an actual changed support.

Define the finite coefficient difference

$$\chi_c(\delta_c) = \big[\mathbf{w}_c(n_{0,c} + \delta_c) - \mathbf{w}_c(n_{0,c})\big]^\top \mathbf{z}_{0,c}. \tag{46}$$

Let w be the reference winner, including the fixed tie rule, and let $g_{wc} = R_c(D_0, x) - R_w(D_0, x) \geq 0$ for $c \neq w$.

Theorem 3 (Finite coefficient-count sensitivity)**.** *For $\delta \in \mathcal{N}_m$, the fixed-statistic scores have paired gap*

$$g^{\mathrm{cnt}}_{wc}(\delta) = g_{wc} - \big[\chi_w(\delta_w) - \chi_c(\delta_c)\big]. \tag{47}$$

Their prediction differs from w if and only if some $c \neq w$ satisfies

$$\chi_w(\delta_w) - \chi_c(\delta_c) > g_{wc}, \quad \text{or equality holds and } c < w. \tag{48}$$

If every $g_{wc} > 0$, define

$$\eta_m(x) = \max_{\delta \in \mathcal{N}_m} \max_{c \neq w} \frac{\chi_w(\delta_w) - \chi_c(\delta_c)}{g_{wc}}. \tag{49}$$

Then $\eta_m \geq 0$. A value $\eta_m < 1$ excludes a change throughout this counterfactual family, and $\eta_m > 1$ implies a strict change for some member. At $\eta_m = 1$, a change occurs exactly when a maximizing equality involves a rival preceding w in the tie order.

Proof. The counterfactual score of class c is $R_c(D_0, x) + \chi_c(\delta_c)$. Subtracting the winner's score proves Equation 47. A rival defeats w precisely when this gap is negative or is zero with higher tie priority. If no rival does so, w remains the selected minimum. If one does, w cannot be selected. This proves both directions of Equation 48. The set $\mathcal{N}_m$ is finite and contains zero, so the maximum in Equation 49 is attained and is nonnegative. Comparing each positive denominator with its numerator gives the three threshold cases. □

For an initially tied winner, Equation 48 remains exact, but the ratio is undefined for a zero gap. The scores and class order determine the answer without adding an offset. Multiplying every score by the same positive constant multiplies both the numerator and denominator of Equation 49 by that constant, so the ratio is scale-invariant. Integer differences also describe parity changes without requiring a continuous extension.

Score sensitivity alone does not imply decision sensitivity. On a fixed component domain $\mathcal{Z}_c$ with span V_c, a coefficient change produces a nonzero score difference for some component vector if and only if it is not orthogonal to V_c. Even then, a label changes only when a paired gap meets Equation 48. A common additive shift preserves gaps, and a common positive multiplier preserves the class ordering.

The full coefficient vector matters. In $a(n)A_c + b(n)B_c$, a constant ratio $a(n)/b(n)$ does not remove the class-dependent multiplier $b(n)$. For example, take $a(n) = b(n) = n$, $(A_0, B_0) = (1/5, 1/5)$, and $(A_1, B_1) = (1/4, 1/4)$. Counts $(2, 2)$ give costs $(4/5, 1)$, whereas $(3, 1)$ give $(6/5, 1/2)$. The winner changes despite the ratio remaining one. This is a fixed-component counterexample with a legal net transfer when class 1 has a mutable record.

I.3 Reference coefficient normalization and uniqueness

Fixing the coefficients at the reference design gives

$$\widehat{R}_c(D, x) = \mathbf{w}_c(n_{0,c})^\top \mathbf{z}_c(D, x). \tag{50}$$

Proposition 3 (Preservation and explicit count invariance)**.** *The normalized rule preserves every reference class score and tie, for arbitrary positive initial counts. It also agrees with the original score after every replacement preserving those counts. At fixed component statistics, all its coefficient-count differences are zero.*

Proof. At D_0, both rules use $\mathbf{w}_c(n_{0,c})$ and $\mathbf{z}_{0,c}$. If a replacement preserves $n_c = n_{0,c}$, the two rules still use the same coefficient and the same, possibly changed, component vector. Finally, Equation 50 has no current-count argument in its coefficient. These are score identities, so they preserve ties as well as strict comparisons. □

For the distance rule, the components can be chosen as

$$\mathbf{w}_c(n) = \big(L(n), \lambda\big)^\top, \qquad \mathbf{z}_c(D, x) = \big(T(p_c), d_c T(q_c)\big)^\top. \tag{51}$$

Then $\widehat{R}_c = L(n_{0,c})T(p_c) + \lambda d_c T(q_c)$. When all initial retained counts equal L_*, division by that common positive constant gives $\widetilde{R}_c$ in Equation 7. With mixed initial parities, each class instead retains its own $L(n_{0,c})$. Substituting a common retained count would not generally preserve the original predictor. The same argument applies to any predetermined symmetric trimming policy.

For the cache, write $b_c = 100u^\top t_c$ and $\bar{k}_c(D)$ for the current class kernel mean. In cost form, $R_c = -S_c$ has coefficients $(1, -\alpha n)^\top$ and components $(-b_c, \bar{k}_c(D))^\top$. Equation 50 is exactly $-\widetilde{S}_c$ from Equation 8. The initial counts remain fixed throughout an attack, including when the reference design is unequal or has mixed parity.

Proposition 4 (Uniqueness on a component domain)**.** *Consider linear candidate scores* $\mathbf{b}_c(n)^\top \mathbf{z}$ *on the same fixed domain* $\mathcal{Z}_c$ *at every considered count. Require reference-score preservation and count invariance for every* $\mathbf{z} \in \mathcal{Z}_c$*. With* $V_c = \operatorname{span}(\mathcal{Z}_c)$*, these requirements hold if and only if*

$$\mathbf{b}_c(n) - \mathbf{w}_c(n_{0,c}) \in V_c^{\perp} \quad \textit{for every considered } n. \tag{52}$$

In particular, the coefficients are unique when $V_c = \mathbb{R}^{J_c}$*.*

Proof. Reference preservation makes $[\mathbf{b}_c(n_{0,c}) - \mathbf{w}_c(n_{0,c})]^\top \mathbf{z} = 0$ on $\mathcal{Z}_c$. Count invariance makes $[\mathbf{b}_c(n) - \mathbf{b}_c(n_{0,c})]^\top \mathbf{z} = 0$ on the same domain. Adding gives Equation 52, because a vector annihilates a set if and only if it annihilates its linear span. Conversely, that orthogonality makes every candidate score equal to $\mathbf{w}_c(n_{0,c})^\top \mathbf{z}$ on the domain, proving both requirements. A full span has zero orthogonal complement. □

The proposition determines only the projection onto V_c when the span is smaller. It does not establish coefficient uniqueness from labels on finitely many queries, or from comparisons on different count-specific component domains. Nor does it identify a representation-independent property of an arbitrary black-box score. The component definitions and the reference counts are part of the rule being analyzed.

I.4 Exact changes under a real replacement

For a real change $D_0 \mapsto D_1$, fix a class and abbreviate $\mathbf{w}_a = \mathbf{w}_c(n_{a,c})$, $\mathbf{z}_a = \mathbf{z}_c(D_a, x)$ for $a \in \{0, 1\}$. Let $\Delta\mathbf{w} = \mathbf{w}_1 - \mathbf{w}_0$ and $\Delta\mathbf{z} = \mathbf{z}_1 - \mathbf{z}_0$. Expanding the bilinear score gives the exact identity

$$R_c(D_1, x) - R_c(D_0, x) = \underbrace{\Delta\mathbf{w}^\top \mathbf{z}_0}_{\text{coefficient change}} + \underbrace{\mathbf{w}_0^\top \Delta\mathbf{z}}_{\text{statistic change}} + \underbrace{\Delta\mathbf{w}^\top \Delta\mathbf{z}}_{\text{interaction}}. \tag{53}$$

Indeed, substitute $\mathbf{w}_1 = \mathbf{w}_0 + \Delta\mathbf{w}$ and $\mathbf{z}_1 = \mathbf{z}_0 + \Delta\mathbf{z}$, distribute the product, and cancel $\mathbf{w}_0^\top \mathbf{z}_0$. By Equation 50, the normalized change is exactly $\mathbf{w}_0^\top \Delta\mathbf{z}$. Fixing the reference coefficients removes the other two terms from this expansion, not the remaining statistic change.

The four scores $R_c^{ab} = \mathbf{w}_a^\top \mathbf{z}_b$ identify the compared objects. R_c^{00} is the reference score, R_c^{10} changes coefficients alone, R_c^{01} is the normalized score on the actual changed support, and R_c^{11} is the original score on that same support. Thus R_c^{01} is directly evaluable, whereas R_c^{10} need not have a realizable support. Equation 53 is an algebraic identity, not an additive decomposition of prediction harm or a causal attribution.

I.5 Remaining statistics and the limits of invariance

The remaining changes admit the usual interval calculation. Suppose valid componentwise bounds $\boldsymbol{\ell}_{z,c} \le \mathbf{z}_c(D', x) \le \mathbf{h}_{z,c}$ are available at each legal removal and insertion profile. For $\mathbf{w}_{0,c} = \mathbf{w}_c(n_{0,c})$, let $\mathbf{w}_{0,c}^+ = \max(\mathbf{w}_{0,c}, 0)$ and $\mathbf{w}_{0,c}^- = \min(\mathbf{w}_{0,c}, 0)$ componentwise. Then

$$\begin{aligned} \ell_{R,c} &= (\mathbf{w}_{0,c}^+)^\top \boldsymbol{\ell}_{z,c} + (\mathbf{w}_{0,c}^-)^\top \mathbf{h}_{z,c}, \\ h_{R,c} &= (\mathbf{w}_{0,c}^+)^\top \mathbf{h}_{z,c} + (\mathbf{w}_{0,c}^-)^\top \boldsymbol{\ell}_{z,c} \end{aligned} \tag{54}$$

bound the complete normalized score. Each nonnegative coefficient preserves the component inequalities, and each negative coefficient reverses them. Summing proves the bounds. For a shared-budget profile, $\widehat{R}_c - \widehat{R}_w \ge \ell_{R,c} - h_{R,w}$. A strictly positive minimum over all rivals and all profiles extendable to a legal global replacement therefore certifies the prediction. Component extrema need not be jointly attainable, so these intervals can be conservative. The distance and cache source-level bounds provide the required component bounds.

Reference preservation is conditional on the fixed design snapshot. Recording a new reference after contamination does not recover an earlier predictor. More generally, suppose an original predictor takes different labels on two copy-related supports. A single modified predictor cannot both agree with the original on every support and be invariant between every such pair: agreement on these two supports would already contradict invariance.

Complete invariance to duplication has a further consequence when required on all finite labeled multisets. If adding any already present record leaves a function unchanged, the function depends only on the distinct-record support. To prove this, begin with one copy of each distinct record and add duplicates until any prescribed positive multiplicities are reached. Invariance at each step makes every such multiset have the same output. This statement concerns unrestricted duplicate additions, not just fixed-total replacements. Reference coefficient normalization makes neither requirement: it preserves the initial score while leaving empirical-frequency effects inside the component statistics.

J Exact Gaps for One Original-Source Cache Copy

The cache permits an exact calculation over the finite one-copy family. Unlike the bounded-key intervals, this calculation uses only kernels of original support records. It separates the effect of their absolute values in the original sum from their within-class deviations after normalization.

J.1 Attack family and mean updates

Fix a query and a reference cache with $n_c > 1$ records per class. In this section, n_c always denotes the initial count. For original source i of class c, define

$$k_{ci} = \exp[-\beta(1 - u^\top z_i)], \qquad \mu_c = \frac{1}{n_c} \sum_{i:y_i=c} k_{ci}, \quad h_c = \max_{i:y_i=c} k_{ci}, \quad l_c = \min_{i:y_i=c} k_{ci}. \tag{55}$$

The source features satisfy the fixed cache interface, so $0 < e^{-2\beta} \le k_{ci} \le 1$. Let $b_c = 100u^\top t_c$ and $\alpha > 0$. For a changed cache D', with current count n'_c, the two scores are

$$S_c(D') = b_c + \alpha \sum_{i \in D'_c} k_i, \qquad \widetilde{S}_c(D') = b_c + \alpha \frac{n_c}{n'_c} \sum_{i \in D'_c} k_i. \tag{56}$$

Both equal $S_c = b_c + \alpha n_c \mu_c$ at the reference cache. They select the largest score. Let w be their common clean winner and $g_{wc} = S_w - S_c \ge 0$.

Let $\mathcal{F}_U$ contain the unchanged cache and every replacement of one mutable position by a complete original key and its correct label from a different class. Donors may be protected, but the inserted copy remains in a mutable position. An operation denoted $a \to b$ removes a record labeled a and copies an original record labeled b. Both records are selected from the original cache, and every class remains nonempty because $n_c > 1$. Let $\mathcal{F}_0$ denote C0, in which every position is mutable.

Deleting or inserting a kernel k changes a class mean by

$$\frac{n\mu - k}{n-1} - \mu = \frac{\mu - k}{n-1}, \qquad \frac{n\mu + k}{n+1} - \mu = \frac{k - \mu}{n+1} \tag{57}$$

for deletion and insertion, respectively. Multiplication by the fixed mass αn gives the corresponding normalized score change.

J.2 Proof of the exact C0 theorem

Use the four nonnegative deviations A_w, B_c, C_c, D_w in Equation 9. Here C_c and D_w are scalar gap changes, not protected sets or support multisets. We prove both minima in Equation 10, including their attainability.

Proof of Theorem 1. For the original rule, an operation $w \to c$ deleting kernel k_{wi} and copying kernel k_{cj} decreases the gap by $\alpha(k_{wi} + k_{cj})$. These choices are independent across the two classes, so its maximum is $\alpha(h_w + h_c)$ and is attained. A move from w to another class decreases the gap by at most αh_w, and a move from another class to c by at most αh_c. Reverse moves increase the gap because kernels are positive, and moves outside the pair leave it unchanged. Thus none is worse than the displayed attainable transfer, proving the original-score formula.

For normalization, Equation 57 shows that $w \to c$ has maximum gap decrease $A_w + B_c$, attained by the two class maxima. A move $c \to w$ has maximum decrease $C_c + D_w$, attained by the two class minima. A move involving just one member of the pair contributes at most A_w, B_c, C_c, or

D_w, according to its role. An outside move or no edit contributes zero. Since all four quantities are nonnegative, each paired sum dominates its two single terms. The two paired directions are feasible even with only two classes. Their larger attainable decrease is therefore the exact maximum over the whole family, proving the normalized-score formula. □

For either rule, let $g_{wc}^{\min}$ denote its corresponding minimum. A one-copy change of prediction exists if and only if some rival has $g_{wc}^{\min} < 0$, or $g_{wc}^{\min} = 0$ and $c < w$. A minimizing copy exists because the family is finite. That rival then defeats w, even if a third class ultimately wins. Conversely, every changed prediction has a rival that defeats w under the same comparison. The minimizing copies may differ across rivals. The existence statement does not require one copy to attain all minima simultaneously.

J.3 Consequences and the scope of comparison

Holding the kernel means fixed, a unit transfer $w \to c$ decreases the original gap by $\alpha(\mu_w + \mu_c)$. Any other unit transfer decreases it by at most this amount. Since $h_w \geq \mu_w$ and $h_c \geq \mu_c$, a strict fixed-mean count flip under C0 implies a realizable one-copy flip. An adverse tie also transfers to a realizable flip, either through a strict inequality or the same tie priority. This implication uses positive kernels and unrestricted deletion of the original winner maximum.

If each class has a constant kernel value, all four deviations in Equation 9 vanish. Normalization then preserves every class score throughout the one-copy family. The original gap can still decrease by $\alpha(\mu_w + \mu_c)$, distinguishing kernel levels from within-class variation.

Proposition 5 (C0 comparison over the complete one-copy family)**.** *Under the assumptions of Theorem 1, the normalized minimum gap is at least the original minimum gap for every rival. Consequently, every prediction stable under the original rule throughout $\mathcal{F}_0$ is also stable under the normalized rule throughout that family, including the fixed tie order.*

Proof. Nonnegative kernels imply $n\mu \geq h$. Hence $n(h-\mu)/(n-1) \leq h$, bounding $A_w \leq \alpha h_w$. Removing an occurrence of the minimum also gives $n\mu - l \leq (n-1)h$, so $n(\mu - l)/(n-1) \leq h - l \leq h$, bounding $C_c \leq \alpha h_c$. The insertion terms satisfy $B_c \leq \alpha h_c$ and $D_w \leq \alpha h_w$, since $l \leq \mu \leq h$ and $n/(n+1) < 1$. Thus each paired sum in Equation 10 is at most $\alpha(h_w + h_c)$. Subtraction from the same clean gap proves the comparison. A nonnegative original minimum stays nonnegative, and any strict inequality required by an adverse tie priority stays strict. The exact prediction criterion proves stability inclusion. □

This comparison is over the complete C0 family, not the scores under an identical chosen edit. A reverse move can decrease a normalized gap while increasing the original gap. That observation does not contradict the comparison of their separate worst cases. The proposition does not compare protected-source families, multiple replacements, arbitrary new keys, or the sufficient interval certificates at those larger threats.

J.4 Allowing same-class copies

A same-class copy keeps all counts fixed but can change the class kernel mean. Replacing kernel k_i by k_j changes both cache scores by the same amount, $\alpha(k_j - k_i)$. Thus the exact normalized cross-class gap in Equation 10 does not alone characterize a family that also permits same-class donors.

Under the C0 assumptions, allowing both kinds of copy gives normalized maximum gap loss

$$E_{wc}^{\text{all}} = \max\{A_w + B_c, C_c + D_w, \alpha(h_w - l_w), \alpha(h_c - l_c)\}. \tag{58}$$

The two additional losses are attained by replacing a winner maximum with its minimum or a rival minimum with its maximum. Other same-class copies cannot decrease this pairwise gap further. The original maximum loss remains $\alpha(h_w + h_c)$: it bounds both within-class ranges as well as all cross-class moves. Each term in Equation 58 is bounded by this original loss, so C0 stability inclusion also holds for the enlarged family. Its normalized stable set can nevertheless be smaller than for cross-class copies alone. The empirical counts in Section 5.5 use the declared cross-class family.

J.5 Protected positions and feasible one-sided moves

Let h_c^U and l_c^U be the maximum and minimum kernels among mutable positions in class c, whenever U_c is nonempty. Donor extrema h_c, l_c still use all original records. Replace A_w, C_c by

$$A_w^U = \frac{\alpha n_w}{n_w - 1}(h_w^U - \mu_w), \qquad C_c^U = \frac{\alpha n_c}{n_c - 1}(\mu_c - l_c^U), \tag{59}$$

and leave B_c, D_w unchanged. The deletion quantities can now be negative.

Proposition 6 (Exact protected-source gaps)**.** *Suppose every class has a mutable record and $K \geq 3$, as in the five-class C4 design. Then*

$$\min_{D' \in \mathcal{F}_U} [S_w(D') - S_c(D')] = g_{wc} - \alpha(h_w^U + h_c), \tag{60}$$

$$\min_{D' \in \mathcal{F}_U} [\widetilde{S}_w(D') - \widetilde{S}_c(D')] = g_{wc} - E_{wc}^U,$$

$$E_{wc}^U = \max\{A_w^U + B_c, C_c^U + D_w, A_w^U, B_c, C_c^U, D_w, 0\}. \tag{61}$$

Proof. The original-score proof uses the largest removable winner kernel, h_w^U, and the largest original rival donor, h_c. Their joint transfer is feasible and dominates all other decreases by positivity, proving Equation 60.

For normalization, the paired transfers attain $A_w^U + B_c$ and $C_c^U + D_w$ by Equation 57. For any outside class o, moves $w \to o$, $o \to c$, $c \to o$, and $o \to w$ attain A_w^U, B_c, C_c^U, and D_w, respectively. Outside donors exist, and the assumed mutable positions make the required outside deletions legal. The unchanged cache gives zero. These cases exhaust all class roles of a cross-class replacement, and the selected extrema maximize each case. Taking the maximum proves Equation 61. Negative deletion terms prevent the C0 simplification to two paired sums. □

With more restrictive permissions, the same proof gives an exact formula after omitting infeasible terms. Include $A_w^U + B_c$ only if $U_w \neq \varnothing$ and $C_c^U + D_w$ only if $U_c \neq \varnothing$. Include A_w^U or C_c^U alone only if its deletion is legal and an outside class exists. Include B_c or D_w alone only if some outside class has a mutable record. Always include zero. Undefined mutable extrema are never evaluated. For the original score, the general maximum decrease is the maximum of zero, $\alpha(h_w^U + h_c)$ when $U_w \neq \varnothing$, and αh_c when an outside mutable class exists. The other feasible decreases are dominated by these terms or are nonpositive.

The C0 comparison can fail under C4. For example, take $\alpha = 1$ and ten records per class, four protected. Let the winner's protected kernels be one and its six mutable kernels be $1/7$, with every rival kernel equal to $1/7$. These values lie in $[e^{-2}, 1]$. The original maximum decrease against any rival is $2/7$. In contrast, inserting a low-kernel winner donor after removing a rival record gives $C_c^U + D_w = 24/77$. A fixed text-score difference can set one clean gap to $3/10$, strictly between these values, and all other gaps larger. The original rule is then stable throughout the one-copy family, whereas normalization can flip. This example locates the role of source permissions without changing the C0 result.

J.6 Relation to the larger threat

All minima in this section are per query over at most one copy from the finite original donor pool, including no edit. They do not describe a single jointly worst cache for multiple queries. The arbitrary bounded-key source-replacement certificate in Appendix H.3 retains its larger intervention set and budgets through four replacements. Exactness for the finite one-copy family neither replaces that certificate nor changes its reported outcomes. Appendix K reports the numerical checks and saved-feature analysis.

K Count and Copy Analysis on Saved Features

K.1 Scope and evaluation units

The supplementary analysis tests the general correction and exact copy formulas on the already evaluated support sets. It is post hoc: no new inputs, encoder inference, model fitting, or coefficient

selection are introduced. The distance analysis reuses 150 model, task, support-set, and permission units from Section D. Applying three fixed trimming policies gives 450 scoring units. The cache analysis reuses 160 units, consisting of two encoders, two tasks, twenty episodes per task, and two source-permission conditions. The resulting 610 analysis units share inputs and are not 610 independent samples.

Three predetermined aggregation policies separate parity from more general retained-count variation. They remove $\lfloor (n-1)/2 \rfloor$, $\lfloor n/4 \rfloor$, or zero values from each end, respectively. The first is the main distance rule; the second retains approximately the central half; the third uses every value. All use the same component weight $\lambda = 25$ and their own initial retained counts in Equation 6. Their comparison is a mechanism analysis, not a search for a replacement primary method. The cache retains $\alpha = 2$ and $\beta = 1$.

Each unit enumerates the complete one-record, cross-class original-source copy family. The actual changed score is compared with a coefficient-only counterfactual, which keeps the original statistics, and the corrected score, which keeps the initial coefficients. The finite-difference identity in Appendix I is evaluated for every candidate and query. Across policies and permissions there are 1,177,600 candidate states and 16,499,200 candidate-query evaluations. Repeated states and inputs are counted repeatedly in these computational totals.

K.2 Coefficient and statistic changes

The numerical decomposition agrees with direct score evaluation. Its largest absolute residual is 1.910×10^{-14} in float64. The initially computed predictions and the primary single-copy harm counts agree with the saved main-study results. Table 17 reports the complete condition-level analysis, including cases in which coefficient-only and actual harms differ. These counts are not an additive attribution of accuracy loss: coefficient, statistic, and interaction terms can reinforce or offset one another in a single margin.

The cache provides a stronger implication within C0. Of 325 initially correct predictions vulnerable to actual copies, 315 are also vulnerable when class means are fixed and only counts change. The full contingency table has 375 initially correct pairs vulnerable to neither intervention, ten vulnerable only to actual copies, none vulnerable only to the count counterfactual, and 315 vulnerable to both. This nesting follows from the positive-kernel maximum bound in Appendix J. The remaining ten cases demonstrate that fixed-mean count changes do not describe every actual copy failure. The implication is specific to the stated cache and permissions.

Table 17: Complete reused-feature mechanism analysis. Count-only harm holds component statistics fixed. Copy harm uses actual source copies. Counts refer to initially correct query-model pairs, not independent samples.

Rule	Task	Sources	Pairs	Clean	Count	Old harm	New harm
Cache	CIFAR-FS	C0	400	393	68	75	1
Cache	EuroSAT	C0	400	307	247	250	16
Cache	CIFAR-FS	C4	400	393	68	75	1
Cache	EuroSAT	C4	400	307	247	250	15
Quarter trim	CIFAR-FS	C0	150	137	7	15	2
Quarter trim	EuroSAT	C0	150	78	16	30	10
Maximal	CIFAR-FS	C0	150	138	8	10	6
Maximal	EuroSAT	C0	150	65	15	21	12
Untrimmed	CIFAR-FS	C0	150	140	6	13	4
Untrimmed	EuroSAT	C0	150	84	13	27	9
Quarter trim	CIFAR-FS	C4	150	137	7	14	1
Quarter trim	EuroSAT	C4	150	78	16	28	9
Maximal	CIFAR-FS	C4	150	138	8	10	6
Maximal	EuroSAT	C4	150	65	15	21	12
Untrimmed	CIFAR-FS	C4	150	140	6	13	3
Untrimmed	EuroSAT	C4	150	84	13	24	8

Table 17 (continued)

Rule	Task	Sources	Pairs	Clean	Count	Old harm	New harm
Quarter trim	AG News	C4	320	188	10	17	10
Quarter trim	Banking77	C4	320	274	3	7	4
Quarter trim	Emotion	C4	320	197	22	32	22
Maximal	AG News	C4	320	163	19	27	9
Maximal	Banking77	C4	320	265	17	18	9
Maximal	Emotion	C4	320	167	25	35	12
Untrimmed	AG News	C4	320	197	20	31	15
Untrimmed	Banking77	C4	320	279	2	4	2
Untrimmed	Emotion	C4	320	210	31	63	26

K.3 Initial designs with mixed counts

The general correction also preserves the score of an initial design whose class counts differ in parity. To check this identity, we form a separate reference design by replacing the first mutable class-0 record with the first class-1 donor. This yields a deliberately mixed count vector. We compute its original score directly from the retained sums or kernel sums, then fix the coefficients at that design's own counts. Across all 610 analysis units, prediction discrepancies are zero and the maximum score difference is 7.105×10^{-15}.

This check changes the declared initial design rather than updating a defense during an attack. Reusing the previous design's common coefficient on the newly declared design need not preserve its predictions. Conversely, after a design is fixed for deployment, its reference counts remain fixed throughout all source interventions. The experiment distinguishes these two operations and tests exact score preservation, not invariance to arbitrary support-content changes.

K.4 Exact margins and exhaustive copy evaluation

The closed-form cache margins agree with direct enumeration in every saved cache unit. Across the 160 C0 and C4 units, the maximum pairwise margin discrepancy is 2.220×10^{-14}, with zero disagreement in copy-induced prediction changes. All 2,000 C0 or 1,200 C4 candidates per query are included. The finite family also includes no intervention when counting robust-correct predictions, so an initially wrong prediction cannot count as robust-correct.

A separate scalar implementation checks the algebra without the feature-loading or score-decomposition code. It directly copies arrays of positive kernels and class labels in 200 constructed instances, covering 2,600 winner-rival pairs. The largest formula-versus-enumeration difference is 3.553×10^{-15}. The reverse class-transfer direction is strictly worse in 1,319 pairs; an outside-class single change is necessary in 76 protected-source pairs. These checks support both terms of Equation 10 and the additional terms under protected permissions. They are numerical verification of a proved finite-family identity, not an independent model experiment or a formal floating-point guarantee.

An exact-rational check tests the C0 nondegradation consequence without numerical tolerance. Enumerating 5,456 scalar arrays verifies all four deviation bounds, and directly copying 1,296 positive-kernel class pairs reproduces the exact gaps and their ordering. The same implementation verifies the protected-source counterexample with original loss $2/7$, normalized loss $24/77$, and gap $3/10$. These algebra checks supplement the general proof; they introduce no new empirical claim.

K.5 Adaptation utility under a common finite threat

The robust-correct count compares all classifiers under the same original-donor, cross-class single-copy family. For a rule f, it counts queries with correct initial prediction and $f_{D'}(x) = f_D(x)$ for every permitted D', including $D' = D$. Each query may have a different worst copy; the count is not accuracy after one shared poisoned cache. The full-support prototype averages the current class features and normalizes each resulting class representative before comparison with the query. Its clean-correct total is 695, compared with 700 for both caches and 643 for zero-shot.

Table 18: Complete cross-class single-copy robust-correct counts. Each task uses 400 query-model pairs. C4 repeats the C0 inputs with four protected sources per class. Protected prototypes are defined only in C4 and remain unchanged under its permitted copies.

Sources	Task	Original	Normalized	Prototype	Zero-shot	Protected
C0	CIFAR-FS	318	392	373	391	–
C0	EuroSAT	57	291	264	252	–
C4	CIFAR-FS	318	392	377	391	378
C4	EuroSAT	57	292	272	252	277

Prototype robustness is computed by exhaustive copies, while the normalized cache is evaluated both analytically and by enumeration.

The normalized cache preserves useful predictions that the fixed text reference lacks. In C0 it has 67 correct predictions that zero-shot misses, while zero-shot has ten correct predictions that the cache misses: one on CIFAR-FS and nine on EuroSAT. Of those 67 additions, 56 remain correct under every cross-class stored-source copy; the original cache retains none. CIFAR-FS contributes three retained additions and EuroSAT contributes 53. Table 18 retains both task-level results, and C4 yields 684 robust-correct predictions in total, compared with 649 for the full-support prototype, 643 for zero-shot, and 655 for the protected prototype.

Post hoc uncertainty uses the original forty support episodes as clusters. We resample twenty episodes within each task, with the same multiplicities for both encoders, and compute 20,000 paired bootstrap replicates. In C0, normalized-cache gains over zero-shot and full-support prototypes are 5.00 and 5.75 percentage points, with descriptive 95% intervals $[3.500, 6.500]$ and $[2.125, 9.625]$. The C4 gains are 5.125, 4.375, and 3.625 points over zero-shot, full-support prototypes, and protected prototypes, respectively, with intervals $[3.625, 6.625]$, $[0.750, 8.125]$, and $[0.375, 7.000]$. These supplementary intervals are unadjusted for multiple comparisons; they do not replace the prespecified simultaneous intervals in Appendix H.7.

The broader certificate answers a different robustness question. Among the 67 cache-correct, zero-shot-wrong predictions, the arbitrary bounded-feature certificate retains twelve at budget one, comprising one on CIFAR-FS and eleven on EuroSAT, and none at budgets two through four. All 226 normalized-cache certificates at budget four concern predictions that zero-shot already gets right. Thus the exact-copy analysis demonstrates retained adaptation utility within a specified stored-source threat, while the arbitrary-replacement analysis demonstrates a paired certificate improvement without additional certified utility over zero-shot at the primary budget.

L Earlier Evidence and the Fresh Confirmation

Earlier support panels provide a separate comparison on the samples used during method development. We recompute them with the same feature pipeline as the fresh panels. They share encoder families and support sizes but use different sources; the earlier Banking77 task also uses different fixed intents. We report these panels separately and do not pool them into the fresh bootstrap analysis.

The earlier text panel has 442 original and 479 normalized certified-correct predictions, with 483 under fixed counts. Thus the same-family gap is 41 and the gain 37. Earlier image counts are 126, 132, and 133, respectively, giving a gap of seven and a gain of six. The earlier SigLIP result has a zero gap and zero gain. By contrast, fresh SigLIP has a gap of one and gains one, while fresh CLIP-B/16 has a gap of one and gains zero. These paired observations distinguish an upper bound from a prediction of realized improvement.

The earlier text panel gains 60 stability certificates and loses none at the primary budget. Stability counts include initially incorrect predictions, unlike certified-correct counts. The fresh panel also has no certified-correct losses at that budget. These observations describe the evaluated supports; Proposition 2 establishes the fixed-count bound without requiring universal dominance between the two variable-count rules.

Table 19: Earlier paired distance results, kept separate from the fresh confirmation panel. Certified counts use budget four. Historical text copy harm uses its original recorded copy pool and is not pooled with the fresh cross-class protocol.

Encoder	N	Clean	Certified old/new	Harm old/new
Qwen2.5-0.5B	192	102	52/56	26/21
Qwen3-0.6B	192	94	55/58	13/8
Llama-3.1-8B	192	150	116/125	14/7
Gemma2-2B	192	150	131/135	11/7
Qwen3-Embedding	192	140	88/105	35/10
CLIP-B/32	100	59	36/37	9/4
CLIP-B/16	100	64	36/41	8/4
SigLIP-B/16	100	69	54/54	4/1

The fresh study improves the separation between method selection and confirmation. The scoring rule, source selection, primary budget, copy family, and fixed-count analysis were fixed before the new feature readouts. Earlier fixed-count gap explanations arose after their corresponding results and remain developmental evidence. Both sets support the same mechanism, but only the fresh study evaluates the complete prespecified confirmation criterion reported in the main text.

M Reproduction and Result Interpretation

M.1 Reproduction order

Reproduction begins with `reproduce_colab.py` and its accompanying `README.md`. Pasting the complete script into one Google Colab cell downloads the public datasets and model weights, reconstructs the specified support and query inputs, and computes their features. Model and dataset revisions, source indices, labels, prompts, and augmentation seeds specify the experimental protocol. No stored features or result values are required as inputs.

The analysis constructs fixed preprocessing and anchors for each support set, then evaluates all six mutable-support distance scores. It computes scalar replacement intervals and fixed-count and variable-count margins for budgets zero through four. Protected and class-text references use the same queries. Exhaustive enumeration supplies the finite-copy results. The cache stage evaluates both rules and the two shared attack selectors, including the held-out split and saved prefixes. Aggregation retains every model, task, permission condition, and source cluster.

The script exports `fewshot_results.zip`, containing `protocol.json`, source-selection records, compact readouts, per-unit results, summaries, and numerical tables. The `confirmation/`, `earlier/`, `cache/`, and `mechanism/` directories separate the four analysis panels. The `analysis/` directory contains residual certificate profiles, timing records, and mathematical checks. All five figures are generated as PNG files. The first illustrates the explicit construction; the other four display measured results. The reported numbers use this public-input computation throughout.

The same entry point performs the post hoc count analysis on the computed features. It enumerates the three distance policies and cache copies, evaluates the coefficient/statistic decomposition and common-threat prototypes, and computes paired intervals. Separate scalar-array and exact-rational routines check the copy formulas, C0 nondegradation, and the C4 counterexample. These analyses share the existing support sets and do not add independent model samples. Completed units and compact readouts support resumption without repeating finished encoder work.