BEYOND SAMPLE RELIABILITY: RELATION IDENTITY DETERMINES NOISE HARM IN SUPERVISED CONTRASTIVE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

In supervised contrastive learning (SupCon), noisy labels alter not only individual supervision but also cross-sample positive relations. We ask which of these relations should be trusted. Through controlled noise placement and quantity-matched interventions, we find that sample difficulty can help identify corrupted supervision but does not consistently predict its representation harm, while changing which relations are modified can produce substantially different outcomes under the same intervention budget. Motivated by this observation, we propose Posterior-Guided Relation Correction (PRC), which learns class posteriors from noisy data and uses their pairwise agreement to reallocate normalized positive target mass among observed-positive relations while preserving instance positives. We further characterize this reweighting at the gradient level: the effect of a relation perturbation depends jointly on its assigned target mass and the current gradient geometry, and posterior-guided reweighting is locally beneficial only when the support score aligns with relation utility. Experiments across synthetic corruption, varying noise rates, and real-world noisy-label benchmarks show consistent representation improvements. Further analysis reveals a clear boundary: strong relation diagnosis does not necessarily support reliable intervention, especially when recovering missing positive relations.

1 INTRODUCTION

Supervised contrastive learning (SupCon) uses class labels to define cross-sample positive relations (Khosla et al., 2020), so the effect of label noise is not confined to a single incorrect supervision target. A wrong label can simultaneously alter multiple training relations: examples from different classes may be incorrectly treated as positives, while true same-class examples may be excluded from the positive set. This raises a basic question: *is the sample itself really the appropriate unit for measuring and intervening on supervision reliability?* Deep networks are known to memorize corrupted labels, and noisy-label learning has therefore developed many sample-level selection, reweighting, and relabeling strategies (Zhang et al., 2016; Arpit et al., 2017; Han et al., 2018; Li et al., 2020; Northcutt et al., 2021). Recent methods further refine this sample-level view using difficulty, confidence tracking, gradient control, or transition modeling (Ye et al., 2025; Zhang et al., 2026a; Yuan et al., 2026; Li et al., 2025; Pan et al., 2025b). However, our controlled experiments show that sample difficulty can help localize corrupted supervision without consistently predicting its representation harm, while whole-sample removal also discards useful instance information and still-valid cross-sample relations.

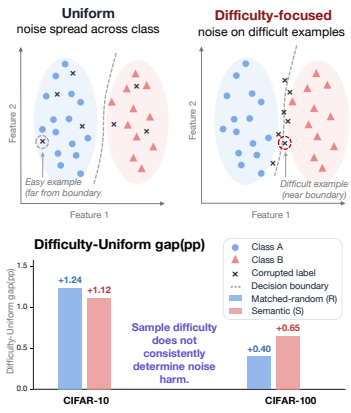


Figure 1: Sample difficulty does not consistently determine noise harm.

Motivated by this observation, we analyze noisy-label SupCon through the relations induced by corrupted labels and use controlled corruption and matched interventions to ask which relation

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.

changes actually affect representation learning. Incorrect labels induce two types of relational discrepancy: false-positive (FP) relations introduce positive supervision that should not exist, whereas false-negative (FN) relations remove true positive supervision. Under quantity-matched interventions, modifying different relations can produce substantially different outcomes even when the number of modified relations is held fixed. This shows that intervention quantity alone is insufficient to characterize the effect of noisy supervision on representation learning.

To obtain a practical intervention without access to clean labels, we propose Posterior-Guided Relation Correction (PRC). Prior noisy-label contrastive methods use reliable examples, pseudo labels, or pair construction to improve robust representation learning (Wu et al., 2021; Li et al., 2022; Yao et al., 2021; Huang et al., 2023; Liu et al., 2025), while broader contrastive-learning work shows that pairwise structure and hard examples can strongly shape representation quality (Chen et al., 2020; Chuang et al., 2020; Xue et al., 2022; Zhang et al., 2026b). PRC instead learns class posteriors using only noisy training data and uses pairwise posterior agreement as semantic support. Rather than removing entire samples or making binary keep-or-remove decisions for relations, PRC redistributes normalized positive target mass among observed-positive relations according to this support, while preserving instance-positive supervision. We further analyze this relation reweighting at the gradient level and show that whether a relation score is beneficial depends not only on its ability to identify suspicious relations, but also on whether it aligns with the local utility of those relations under the current optimization state. Experiments under controlled synthetic corruption and naturally occurring noisy supervision show consistent improvements in representation quality. At the same time, our diagnosis–intervention analysis reveals an important boundary: the ability to identify suspicious relations does not necessarily imply that they can be reliably corrected, particularly when recovering missing positive relations.

Our main contributions are summarized as follows:

1. We conduct controlled relation-corruption analyses that separate the effects of sample difficulty, wrong-target assignment, and relation identity. Quantity-matched interventions further show that modifying different relations can lead to substantially different outcomes even when the intervention size is fixed.
2. We propose Posterior-Guided Relation Correction (PRC), which uses noisy-only posterior semantic support to redistribute normalized positive target mass among observed-positive relations while preserving all training examples and instance-positive supervision.
3. We characterize relation reweighting from a local optimization perspective and distinguish relation diagnosability from intervention actionability. In particular, strong relation-ranking performance does not necessarily translate into reliable intervention for missing positive relations.

2 Related Work

2.1 Learning with Noisy Labels

Deep networks can memorize random or corrupted labels, and this memorization interacts strongly with example difficulty and training dynamics (Zhang et al., 2016; Arpit et al., 2017; Toneva et al., 2018; Liu et al., 2020). Learning with noisy labels has therefore been extensively studied through sample selection, loss reweighting, label correction, transition modeling, and semi-supervised learning (Han et al., 2018; Arazo et al., 2019; Li et al., 2020; Northcutt et al., 2021; Wei et al., 2021). Most existing methods estimate the reliability of individual examples from training loss, prediction confidence, model agreement, temporal consistency, or estimated label-transition structure, and then remove, downweight, or relabel suspected noisy samples (Yao et al., 2021; Huang et al., 2023; Liu et al., 2025; Pan et al., 2025a; Ye et al., 2025; Zhang et al., 2026a; Yuan et al., 2026; Fan & Li, 2025; Li et al., 2025; Pan et al., 2025b; Heck et al., 2025; Liang et al., 2025; Wang et al., 2025; Kim et al., 2025; Staats et al., 2025). While effective for robust classification, these approaches primarily treat reliability as a sample-level property. In supervised contrastive learning (SupCon), however, a single label directly determines multiple cross-sample relations, making sample-level reliability alone insufficient to characterize the resulting supervision errors.

2.2 NOISY-LABEL CONTRASTIVE LEARNING

Recent work has incorporated pairwise and contrastive supervision into noisy-label learning, building on modern contrastive and self-supervised representation objectives (Chen et al., 2020; Khosla et al., 2020; Chuang et al., 2020; He et al., 2020; Grill et al., 2020). Class2Simi converts noisy class labels into pairwise similarity supervision (Wu et al., 2021); Sel-CL constructs confident positive pairs from selected reliable examples (Li et al., 2022); and MOIT, Jo-SRC, TCL, and PLReMix combine sample reliability, pseudo-label refinement, or relation construction with contrastive objectives (Ortego et al., 2021; Yao et al., 2021; Huang et al., 2023; Liu et al., 2025). Other studies show that contrastive representations can improve robustness under label noise, while the role of hard or difficult examples can differ between supervised and unsupervised contrastive regimes (Ghosh & Lan, 2021; Xue et al., 2022; Zhang et al., 2026b). These studies establish that pairwise structure is important under noisy supervision.

Our focus is complementary: rather than introducing pairwise supervision itself, we study which corrupted relations matter under controlled, quantity-matched interventions. This analysis motivates PRC, which uses noisy-only posterior semantic support to redistribute normalized positive target mass among observed-positive relations, and further examines when relation diagnosability can or cannot be translated into reliable intervention.

3 RELATIONAL CHARACTERIZATION OF NOISY SUPERVISED CONTRASTIVE LEARNING

3.1 PROBLEM SETUP

Given a noisy training set $\mathcal{D} = \{(x_i, \widetilde{y}_i)\}_{i=1}^{N}$, where $\widetilde{y}_i$ is the observed label and the ground-truth label y_i is unavailable during training, supervised contrastive learning constructs label-induced positive and negative sets from the observed labels. For an anchor sample i, these sets are $P_i^{\mathrm{lab}} = \{j \neq i : \widetilde{y}_j = \widetilde{y}_i\}$ and $N_i^{\mathrm{lab}} = \{j \neq i : \widetilde{y}_j \neq \widetilde{y}_i\}$. At the same time, different augmented views of the same sample form instance-level positive relations P_i^{inst} that are independent of class labels.

Unlike standard classification, labels in SupCon do not only determine the supervision target for an individual example; they also determine pairwise interactions between that example and many others. Therefore, when $\widetilde{y}_i \neq y_i$, a single incorrect label rewrites multiple supervised relations simultaneously, rather than merely creating one erroneous sample-level target. This suggests that, in noisy-label SupCon, *label-induced relations may be a more appropriate basic unit than the whole sample for analyzing supervision reliability*.

3.2 LABEL-INDUCED RELATIONAL CORRUPTION

To characterize how label noise changes the supervised structure, we define the ground-truth semantic relation matrix and the observed-label relation matrix as $A^{\star}_{ij} = \mathbf{1}[y_i = y_j]$ and $\widetilde{A}_{ij} = \mathbf{1}[\widetilde{y}_i = \widetilde{y}_j]$, respectively. Ideally, $\widetilde{A} = A^{\star}$. When labels are corrupted, two types of structural discrepancy arise.

The first type is a *false-positive (FP) relation*, defined as $\mathcal{E}_{\mathrm{FP}} = \{(i,j) : \widetilde{A}_{ij} = 1,\ A^{\star}_{ij} = 0\}$. It occurs when examples from different true classes are incorrectly constructed as a positive relation due to noisy labels, introducing *spurious attraction* that should not exist.

The second type is a *false-negative (FN) relation*, defined as $\mathcal{E}_{\mathrm{FN}} = \{(i,j) : \widetilde{A}_{ij} = 0,\ A^{\star}_{ij} = 1\}$. It occurs when examples from the same true class are not constructed as a positive relation because their observed labels differ, and instead participate in contrastive competition, creating *spurious repulsion*.

Therefore, an incorrect label usually rewrites multiple supervised relations around the example, rather than merely creating a single erroneous target. More importantly, *whether an example is mislabeled and whether all relations incident to that example are unreliable are not the same question*. Even when an example has an incorrect observed label, it may still retain useful instance information and some reliable relations. This makes it difficult to accurately describe the corrupted supervised structure by simply treating the entire example as reliable or unreliable.

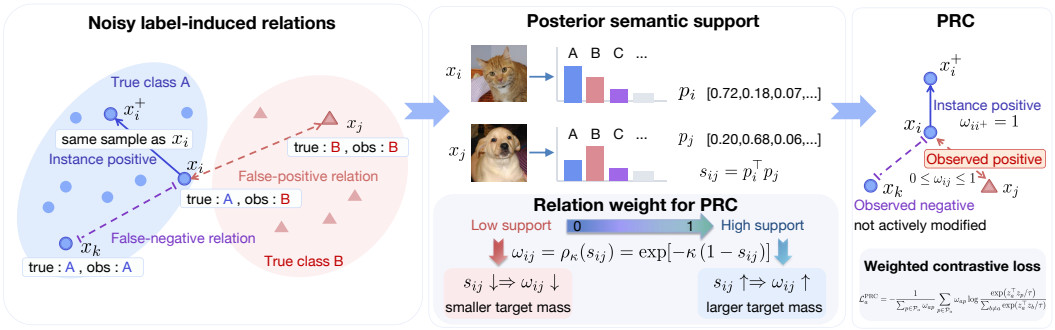


Figure 2: Posterior-Guided Relation Correction (PRC) pipeline. PRC first trains a noisy-only proxy model to estimate class posteriors, converts endpoint posteriors into pairwise semantic support, and uses this support to redistribute normalized positive target mass among observed-positive relations while preserving instance-positive supervision.

3.3 From sample filtering to relation-level intervention

Existing noisy-label methods commonly suppress unreliable supervision through sample filtering. Let $m_i \in \{0, 1\}$ indicate whether sample i is retained, and let $\mathbf{m} = [m_1, \ldots, m_N]^\top$. The corresponding positive relation matrix can be written as $A^{\text{node}} = (\mathbf{m}\mathbf{m}^\top) \odot \widetilde{A}$, where $\odot$ denotes element-wise multiplication. Once sample i is removed, all label-induced positive relations incident to it are deleted simultaneously, regardless of whether those relations themselves are corrupted. Whole-sample filtering is therefore a coarse-grained intervention: it can remove erroneous supervision, but it may also discard relations and instance information that remain useful.

A finer-grained alternative is to assign relation-specific support to label-induced positives and use it to redistribute normalized positive target mass across these relations, while preserving the example and its instance information. Let $W^+ \in [0, 1]^{N \times N}$ denote the positive-relation support matrix. Then $A^{\text{rel}} = W^+ \odot \widetilde{A}$. Unlike the node-level mask $\mathbf{m}\mathbf{m}^\top$, W^+ allows the relations between the same example and different neighbors to be adjusted independently.

Although label noise also creates FN relations, negative-side intervention is not a simple symmetric counterpart of positive-side weighting. Positive-side reweighting changes how normalized positive target mass is allocated within the observed-positive support, whereas modifying observed-negative interactions changes the contrastive competition structure and does not directly restore missing positive supervision. We therefore adopt a conservative design in which the formal relation correction acts on observed-positive relations, and analyze negative-side intervention separately in the experiments.

Thus, the more appropriate question in noisy-label SupCon is no longer only *whether a particular example should be retained*, but rather *which label-induced relations should be trusted and how strongly they should participate in training*. Since the ground-truth relation matrix $A^\star$ is unavailable during training, the next section introduces PRC, which estimates relation reliability from semantic evidence obtained solely from noisy data.

4 Posterior-Guided Relation Correction

Section 3 shows that sample-level removal processes all supervised relations incident to a node jointly, making it difficult to distinguish which relations are truly corrupted by label noise. Based on this observation, we aim to construct fine-grained semantic support for each observed positive relation without relying on clean labels, and then use this support to modulate its role in supervised contrastive learning. PRC does not estimate whether a sample is clean and does not remove training samples. Instead, it directly operates on the cross-sample relations induced by observed labels. The overall procedure consists of a noisy-only proxy posterior, pairwise semantic support, and relation-weighted contrastive learning.

4.1 Posterior Semantic Support

Because the ground-truth relation matrix $A^\star$ is unavailable during training, we first train a proxy classifier h_ϕ using only the noisy training set. For sample x_i, the proxy model produces the class posterior

$$p_i = \mathrm{softmax}(h_\phi(x_i)) \in \mathbb{R}^C, \tag{1}$$

where $p_i(c)$ denotes the posterior probability that sample i belongs to class c. For any sample pair (i, j), we define its posterior semantic support as

$$s_{ij} = p_i^\top p_j = \sum_{c=1}^{C} p_i(c) p_j(c), \qquad s_{ij} \in [0, 1]. \tag{2}$$

A larger s_{ij} indicates stronger agreement between the class-semantic evidence of the two endpoints. We use s_{ij} as a semantic support score, not as a calibrated probability that the relation is correct.

4.2 Posterior-Guided Modulation of Observed Positives

For an anchor view $a = (i, v)$, let $\mathcal{P}_a^{\mathrm{inst}}$ denote the other augmented view of the same sample, and let $\mathcal{P}_a^{\mathrm{lab}}$ denote the cross-sample positive relations constructed from observed labels. For a positive view $p = (j, u)$, we define

$$\omega_{ap} = \begin{cases} 1, & p \in \mathcal{P}_a^{\mathrm{inst}}, \\ \rho_\kappa(s_{ij}), & p \in \mathcal{P}_a^{\mathrm{lab}}, \end{cases} \tag{3}$$

where $\rho_\kappa(s) = \exp[-\kappa(1 - s)]$. The function ρ_κ preserves the ordering of s_{ij} and determines the relative positive-target weight assigned to each observed-positive relation; we use $\kappa = 1$ in the main experiments. This yields the weighted supervised contrastive objective

$$\mathcal{L}_a^{\mathrm{PRC}} = -\frac{1}{\sum_{p \in \mathcal{P}_a} \omega_{ap}} \sum_{p \in \mathcal{P}_a} \omega_{ap} \log \frac{\exp(z_a^\top z_p / \tau)}{\sum_{b \neq a} \exp(z_a^\top z_b / \tau)}, \tag{4}$$

where $\mathcal{P}_a = \mathcal{P}_a^{\mathrm{inst}} \cup \mathcal{P}_a^{\mathrm{lab}}$.

This design follows two principles. First, PRC does not make a binary keep-or-remove decision for label-induced cross-sample positives. Instead, after normalization in Eq. 4, the posterior-derived weights reallocate positive target mass among the observed-positive relations: relations with stronger semantic support receive a larger share of the normalized positive target, while weaker-supported relations receive a smaller share. Second, the instance positive between two augmented views of the same sample always retains unit pre-normalization weight. Thus, PRC preserves instance-level supervision and all training samples while adjusting how positive supervision is distributed across label-induced relations.

PRC only modulates positive relations that have already been proposed by the observed labels; it does not directly convert observed-negative pairs into new positive relations. This is a deliberately conservative intervention: relation support can be used to detect potential missing positives, but diagnostic ability does not necessarily imply that newly added relations have sufficient precision. We further analyze this distinction between FN diagnosis and direct relation recovery in the experiments.

4.3 From Relation Reweighting to Gradient Interference

We analyze the local optimization effect of relation reweighting at a fixed model state. Let t_{ab} and $t_{ab}^\star$ denote the current positive target and the corresponding clean positive target, respectively, and define the target deviation as $\delta_{ab} = t_{ab} - t_{ab}^\star$. Let $J_{ab} = \nabla_\theta(z_a^\top z_b / \tau)$. The gradient deviation induced by the target perturbation is then

$$e_t := g_t - g_\star = -\frac{1}{A} \sum_a \sum_{b \neq a} \delta_{ab} J_{ab}. \tag{5}$$

Thus, even with the same number of corrupted relations or the same amount of target-mass deviation, the resulting optimization effect can differ because the corresponding relation gradients may reinforce

or cancel each other. To characterize whether this deviation interferes with the clean descent direction, we define

$$H_t = -\langle g_\star, e_t \rangle = \|g_\star\|^2 - \langle g_\star, g_t \rangle. \tag{6}$$

When $H_t > 0$, noisy supervision weakens the first-order descent direction of the clean objective at the current state. Here, H_t characterizes only local optimization interference and does not directly imply final generalization performance. We further define the local utility of relation (a, b) as

$$c_{ab} = -\langle g_\star, J_{ab} \rangle. \tag{7}$$

For the PRC reweighted target t_a^κ, we have

$$\frac{d}{d\kappa}\langle g_\star, g_\kappa \rangle = \frac{1}{A}\sum_a \mathrm{Cov}_{t_a^\kappa}\left(\bar{s}_{ap}, c_{ap}\right). \tag{8}$$

Therefore, whether posterior-guided reweighting locally improves the clean descent direction depends not only on corruption ranking, but also on whether semantic support is positively aligned with current relation utility. We empirically examine this local prediction in Appendix D.7. Full derivations and further analyses are provided in the appendix.

4.4 Training Procedure

PRC consists of two training stages. In the first stage, we train the proxy classifier h_ϕ using only the noisy training data and store the class posterior p_i for every training sample. In the second stage, we perform supervised contrastive learning. For each mini-batch, we compute $s_{ij} = p_i^\top p_j$ for cross-sample positive relations constructed by the observed labels, obtain the corresponding relation weight through $\rho_\kappa(s_{ij})$, and optimize the PRC loss above. Instance-level positive relations always retain unit weight. The overall process is summarized as noisy data $\rightarrow \{p_i\} \rightarrow s_{ij} \rightarrow \omega_{ij} \rightarrow \mathcal{L}^{\mathrm{PRC}}$.

Throughout training, PRC never uses clean labels, true noise identities, or oracle difficulty, and it never decides whether a training sample should be kept or deleted as a whole. It only estimates relation-level semantic support and uses it to redistribute normalized positive target mass among the label-induced positive relations associated with each sample.

5 Experiments

5.1 Experimental Setup

We evaluate PRC on controlled synthetic corruption and varying noise rates on CIFAR-10/100, and on real-world noisy-label benchmarks including CIFAR-N, Clothing1M, and Food-101N.

For controlled corruption, sample difficulty is measured by the class-conditional feature margin

$$m_i = z_i^\top \mu_{y_i} - \max_{c \neq y_i} z_i^\top \mu_c, \tag{9}$$

where smaller m_i indicates greater difficulty. Uniform and Boundary differ only in which samples are corrupted. To isolate wrong-target assignment, Semantic selects

$$\tilde{y}_i^S = \arg\max_{b \neq y_i} q_{ib}, \tag{10}$$

while Matched-random permutes the resulting wrong-target multiset within each source class, preserving both the corrupted samples and source–target transition counts.

We verify the difficulty results with CE loss, predictive entropy, and noisy-label EL2N in the appendix. All methods use the same data splits, backbone, augmentation, training budget, and linear-probe protocol. Clean annotations are used only for controlled analysis and post-hoc linear probing, never for training PRC or estimating relation weights; exact real-world probe splits are provided in the appendix.

Table 1: Main comparison under controlled synthetic and real-world label noise. (a) Linear-probe accuracy (%) on CIFAR-10/100 under 40% synthetic corruption. (b) Linear-probe accuracy (%) on real-world noisy-label benchmarks. Results are averaged over three random seeds; full standard deviations for the controlled comparison are reported in Appendix C.4.

(a) Controlled synthetic noise, 40% corruption

Method	Year / Venue	U+R	U+S	B+R	B+S	Avg.
Observed SupCon	NeurIPS 20	81.65 / 54.55	80.77 / 55.52	82.89 / 54.95	81.89 / 56.17	81.80 / 55.30
Sel-CL†	CVPR 22	78.01 / 40.71	76.03 / 41.03	80.02 / 42.14	78.60 / 42.95	78.17 / 41.71
TCL†	CVPR 23	84.65 / 56.60	83.52 / 57.30	84.92 / 57.05	83.55 / 57.93	84.16 / 57.22
Structural Labels†	CVPR 24	83.78 / 58.10	83.45 / 58.92	84.42 / 59.24	84.35 / 59.70	84.00 / 58.99
SV-Learner†	TKDE 24	82.10 / 55.55	81.75 / 56.02	82.94 / 56.66	82.73 / 56.81	82.38 / 56.26
PLReMix†	WACV 25	82.75 / 58.35	82.40 / 58.80	83.58 / 59.18	83.31 / 59.63	83.01 / 58.99
PSSCL†	PR 25	83.62 / 57.72	83.11 / 58.10	84.02 / 58.42	83.71 / 58.86	83.62 / 58.28
DCD†	ICCV 25	83.54 / 51.40	83.11 / 51.85	84.18 / 52.24	84.21 / 52.71	83.76 / 52.05
WSC†	ICML 25	84.58 / 58.18	84.06 / 58.96	84.81 / 59.32	84.56 / 59.74	84.50 / 59.05
CA2C†	ICCV 25	83.84 / 57.56	83.29 / 58.03	84.10 / 58.27	83.92 / 58.65	83.79 / 58.13
JAL-CE†	ICCV 25	83.47 / 57.31	82.98 / 57.69	83.83 / 57.98	83.58 / 58.32	83.47 / 57.83
DSS†	CVPR 26	83.90 / 57.82	83.38 / 58.21	84.54 / 58.82	84.29 / 59.18	84.03 / 58.51
SED+FINE†	CVPR 26	84.02 / 57.91	83.50 / 58.36	84.38 / 58.73	84.12 / 59.04	84.01 / 58.51
NAPCL†	ESWA 26	84.29 / 58.05	83.81 / 58.71	84.66 / 59.06	84.43 / 59.51	84.30 / 58.83
PRC (Ours)	–	**85.18 / 58.92**	**84.36 / 59.87**	**85.54 / 60.11**	**84.86 / 60.65**	**84.99 / 59.89**

(b) Real-world noisy labels

Method	Year / Venue	CIFAR-10N (worse)	CIFAR-100N	Clothing1M	Food-101N
Observed SupCon	NeurIPS 20	79.45 ± 0.38	54.91 ± 0.23	70.84 ± 0.31	83.27 ± 0.24
Structural Labels†	CVPR 24	81.43 ± 0.27	55.92 ± 0.31	72.18 ± 0.25	84.36 ± 0.21
PLReMix†	WACV 25	81.76 ± 0.21	56.14 ± 0.34	72.63 ± 0.18	84.72 ± 0.19
PSSCL†	PR 25	81.58 ± 0.26	56.02 ± 0.32	72.54 ± 0.24	84.61 ± 0.22
DCD†	ICCV 25	81.18 ± 0.30	55.71 ± 0.29	72.31 ± 0.27	84.48 ± 0.23
WSC†	ICML 25	82.08 ± 0.18	56.48 ± 0.27	73.12 ± 0.20	85.06 ± 0.18
CA2C†	ICCV 25	81.71 ± 0.24	56.09 ± 0.31	72.68 ± 0.22	84.73 ± 0.20
JAL-CE†	ICCV 25	81.49 ± 0.29	55.96 ± 0.33	72.45 ± 0.25	84.57 ± 0.24
DSS†	CVPR 26	81.93 ± 0.21	56.31 ± 0.28	72.88 ± 0.21	84.91 ± 0.19
SED+FINE†	CVPR 26	82.01 ± 0.20	56.36 ± 0.30	72.96 ± 0.19	85.02 ± 0.18
NAPCL†	ESWA 26	82.12 ± 0.17	56.52 ± 0.25	73.18 ± 0.18	85.11 ± 0.17
PRC (Ours)	–	$\mathbf{82.58 \pm 0.12}$	$\mathbf{57.03 \pm 0.21}$	$\mathbf{73.71 \pm 0.15}$	$\mathbf{85.61 \pm 0.14}$

(a) reports CIFAR-10/CIFAR-100; U/B: uniform/boundary-focused placement; R/S: matched-random/semantic targets. † Protocol-adapted under our unified setting.

5.2 Main Comparison under Synthetic and Real-World Label Noise

We compare PRC with existing methods under the same data splits, representation-learning budget, and clean-label linear-probe protocol. As shown in Table 1, PRC achieves the strongest average performance under controlled synthetic corruption and the best accuracy across all four real-world noisy-label benchmarks. The consistent improvements across different corruption placements, wrong-target assignments, and naturally occurring noise indicate that the benefit of relation-level modulation is not restricted to a particular noise configuration. To assess whether the unified budget disadvantages schedule-dependent baselines, we additionally restore an extended method-faithful schedule for PSSCL. While this improves PSSCL, PRC under the original 30-epoch budget retains its advantage on both CIFAR-10 and CIFAR-100 (Appendix C.6), indicating that the main comparison is not explained solely by schedule compression. Sensitivity to κ is reported in Table 3; PRC remains stable over a moderate range, and we use $\kappa = 1$ throughout the main experiments.

5.3 Why Posterior Semantic Support?

As shown in Table 2, sample-level posterior weighting improves over unweighted SupCon by $+2.23$ pp on average, whereas relation-level PRC increases the gain to $+3.84$ pp. Importantly, both variants use exactly the same proxy posterior, separating the benefit of additional posterior information from

Table 2: Intervention-granularity and relation-score ablation. Linear-probe accuracy (%); Avg. gain is relative to unweighted SupCon.

Weighting variant	C10 U+R	C10 B+S	C100 U+R	C100 B+S	Avg. gain
No weighting	81.65	81.89	54.55	56.17	–
Sample-level posterior	83.74	83.48	57.12	58.83	+2.23
Random-permuted	82.12	82.08	54.88	56.40	+0.31
Endpoint-loss	84.18	83.62	58.32	59.28	+2.79
Encoder cosine	85.02	84.05	57.42	59.02	+2.81
Binary posterior	84.40	83.88	57.92	59.48	+2.86
Posterior support (PRC)	**85.18**	**84.86**	**58.92**	**60.65**	**+3.84**

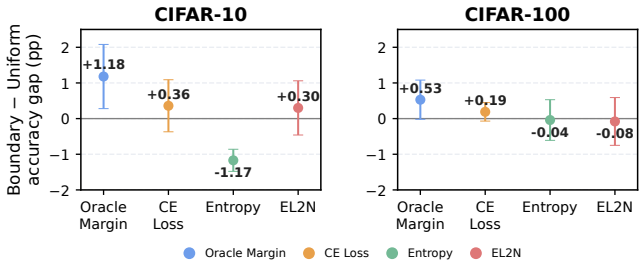


Figure 3: Difficulty-conditioned noise placement.

Table 3: Sensitivity to κ under B+S. Linear-probe accuracy (%).

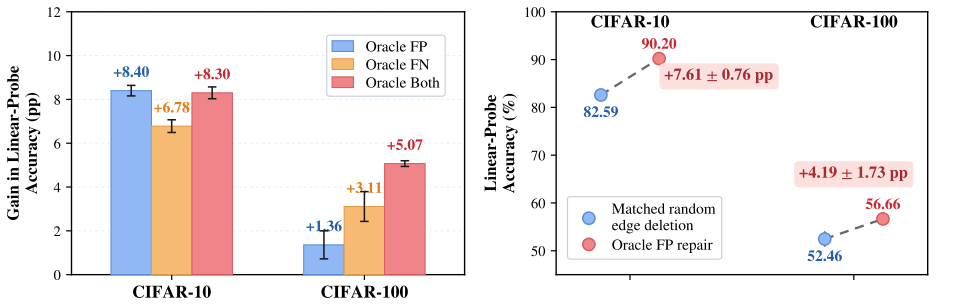
κ	C10	C100
0	81.89	56.17
0.25	82.96	57.82
0.5	83.71	59.18
1.0	**84.86**	**60.65**
2.0	84.08	59.84
4.0	83.62	59.21
8.0	82.91	58.36

that of relation-specific allocation. PRC also remains stable across substantially different proxy-training durations (Appendix D.8). For the sample-level baseline, we compute $q_i = p_i(\widetilde{y}_i)$ for anchor sample i and replace the relation-level weight in Eq. 3 by a shared anchor-level weight $\rho_\kappa(q_i)$ for every $p \in \mathcal{P}_a^{\text{lab}}$; instance positives keep weight 1, and the loss is normalized as in Eq. 4. Posterior support also outperforms endpoint loss, encoder cosine, binary selection, and randomly permuted weights, indicating that the benefit arises from continuous, relation-specific allocation of positive target mass rather than posterior information or heterogeneous weighting alone. A local optimization diagnostic further confirms that posterior support exhibits stronger positive score–utility alignment and produces a larger one-step decrease in the oracle clean objective; see Appendix D.7.

5.4 Sample difficulty does not consistently determine noise harm

Sample difficulty can contain information about contaminated supervision, but this does not mean that it determines the actual harm caused by label noise. We construct boundary-focused corruption using four difficulty definitions oracle margin, CE loss, predictive entropy, and noisy-label EL2N and compare it with uniform corruption matched in sample count and class-transition structure. Notably, some difficulty-related signals can localize erroneous relations well: for example, CE loss and EL2N reach FP-localization AUROC scores of 0.838 and 0.848 on CIFAR-10, respectively. However, the Boundary Uniform gap changes sign across difficulty definitions and datasets, showing that corruption localization does not reliably predict representation harm. Full results are in Appendix Tables 16 and 17.

We further examine whether sample difficulty provides a reliable intervention unit. Once random node deletion is matched to the number of corrupted nodes removed by Drop-hard, the advantage of difficulty-based deletion becomes small or inconsistent. In contrast, PRC retains all samples and performs better in most controlled settings. Full matched node-deletion results are reported in Appendix Figure 5. This suggests that diagnostic usefulness does not imply that the whole sample is the appropriate intervention unit.



(a) Oracle FP/FN relation intervention (gain over Observed) (b) Oracle FP repair vs. matched random edge deletion

Figure 4: Targeted relation repair. (a) Oracle FP, FN, and joint repair gains. (b) Oracle FP repair versus matched random deletion. Error bars show standard deviations; full results are in Tables 18 and 19.

Table 4: Relation diagnosability does not guarantee reliable intervention.

(a) Diagnosis and active FN recovery

Dataset	FP AUROC / AUPRC	FN AUROC / AUPRC	FN Top-5% Prec.	Recovery vs. FP-only	Targeted vs. Random
CIFAR-10	0.846 / 0.890	0.936 / 0.602	0.622	+0.94	+1.01
CIFAR-100	0.727 / 0.797	0.949 / 0.267	0.086	−2.36	+0.02

(b) Conservative FN neutralization: Δ accuracy over FP-only PRC

Dataset	$\eta = .25$	$\eta = .50$	$\eta = .75$	$\eta = 1.0$
CIFAR-10	−0.17	+0.67	+0.22	+0.89
CIFAR-100	+0.18	−1.42	−0.58	−0.22

5.5 Relation Identity Determines the Effect of Corrupted Supervision

To test whether degradation depends on the specific erroneous relations, we use clean labels only for analysis and perform oracle FP/FN interventions. As shown in Figure 4, CIFAR-10 is primarily FP-limited, whereas CIFAR-100 benefits more from FN and two-sided repair. The quantity-matched random-deletion control further shows that the intervention effect is not explained by how many positive relations are removed, but by *which* relations are corrupted and corrected. Since oracle corruption locations are unavailable during training, we next ask whether noisy-data diagnostics can be converted into reliable relation-level interventions.

5.6 Relation Diagnosability Does Not Guarantee Reliable Intervention

Table 4 shows that strong global relation ranking does not necessarily imply reliable intervention. Although posterior disagreement can identify missing-positive relations, the top-ranked candidates may still have insufficient precision, explaining why FN recovery helps on CIFAR-10 but fails on CIFAR-100. A more conservative strategy that suppresses suspected FN interactions also yields inconsistent gains. Thus, noisy-only semantic evidence is useful for diagnosis, but successful intervention requires sufficiently precise relation-level evidence.

6 Conclusion

We study noisy-label SupCon by asking which corrupted relations actually affect representation learning. Controlled and quantity-matched interventions show that sample difficulty does not consistently predict noise harm, while the identity of modified relations matters substantially. Based on this observation, we propose PRC, which uses noisy-only posterior semantic support to redistribute normalized positive target mass among observed-positive relations. Our experiments show consistent representation improvements, while further revealing that relation diagnosability does not necessarily imply reliable intervention, especially for recovering missing positive relations.

7 AI USE STATEMENT

Generative AI tools assisted with code development and debugging, theoretical derivations and proof drafting, literature searches and reference organization, and manuscript drafting, editing, and language polishing. The authors are responsible for the final content, including the correctness of the code, mathematical claims, and reported results.

8 REPRODUCIBILITY STATEMENT

Implementation and evaluation details are provided in Appendix B, with additional ablations and robustness analyses in Appendices C–G. Anonymous code is available at `https://anonymous.4open.science/r/PRC-E6DD/`.

REFERENCES

Eric Arazo, Diego Ortego, Paul Albert, Noel O'Connor, and Kevin McGuinness. Unsupervised label noise modeling and loss correction. In *International conference on machine learning*, pp. 312–321. PMLR, 2019.

Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International conference on machine learning*, pp. 233–242. PMLR, 2017.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. *Advances in neural information processing systems*, 33:8765–8775, 2020.

Wenxiao Fan and Kan Li. Combating semantic contamination in learning with label noise. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 2870–2878, 2025.

Aritra Ghosh and Andrew Lan. Contrastive learning improves model robustness under label noise. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2697–2702. IEEE, 2021.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, 31, 2018.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 9726–9735. IEEE, 2020.

Michael Heck, Christian Geishauser, Nurul Lubis, Carel van Niekerk, Shutong Feng, Hsien-Chin Lin, Benjamin Matthias Ruppik, Renato Vukovic, and Milica Gasic. Learning from noisy labels via self-taught on-the-fly meta loss rescaling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 24059–24068, 2025.

Zhizhong Huang, Junping Zhang, and Hongming Shan. Twin contrastive learning with noisy labels. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11661–11670. IEEE, 2023.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020.

Daehwan Kim, Kwangrok Ryoo, Hansang Cho, and Seungryong Kim. Splitnet: learnable clean-noisy label splitting for learning with noisy labels. *International Journal of Computer Vision*, 133(2): 549–566, 2025.

Jiahui Li, Tai-Wei Chang, Kun Kuang, Ximing Li, Long Chen, and Jun Zhou. Learning causal transition matrix for instance-dependent label noise. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 18305–18313, 2025.

Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394*, 2020.

Shikun Li, Xiaobo Xia, Shiming Ge, and Tongliang Liu. Selective-supervised contrastive learning with noisy labels. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 316–325. IEEE, 2022.

Jiye Liang, Yixiao Li, and Junbiao Cui. Label noise correction via fuzzy learning machine. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 18676–18683, 2025.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

Xiaoyu Liu, Beitong Zhou, Zuogong Yue, and Cheng Cheng. Plremix: Combating noisy labels with pseudo-label relaxed contrastive representation learning. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 6517–6527. IEEE, 2025.

Curtis Northcutt, Lu Jiang, and Isaac Chuang. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70:1373–1411, 2021.

Diego Ortego, Eric Arazo, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Multi-objective interpolation training for robustness to label noise. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6606–6615, 2021.

Bikang Pan, Qun Li, Xiaoying Tang, Wei Huang, Zhen Fang, Feng Liu, Jingya Wang, Jingyi Yu, and Ye Shi. Nlprompt: Noise-label prompt learning for vision-language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19963–19973. IEEE, 2025a.

Weiran Pan, Wei Wei, Feida Zhu, and Yong Deng. Enhanced sample selection with confidence tracking: Identifying correctly labeled yet hard-to-learn samples in noisy data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 19795–19803, 2025b.

Max Staats, Matthias Thamm, and Bernd Rosenow. Enhancing noise-robust losses for large-scale noisy data learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7006–7014, 2025.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. An empirical study of example forgetting during deep neural network learning. *arXiv preprint arXiv:1812.05159*, 2018.

Jialiang Wang, Xianming Liu, Xiong Zhou, Gangfeng Hu, Deming Zhai, Junjun Jiang, and Xiangyang Ji. Joint asymmetric loss for learning with noisy labels. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1947–1956. IEEE, 2025.

Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu. Learning with noisy labels revisited: A study using real-world human annotations. *arXiv preprint arXiv:2110.12088*, 2021.

Songhua Wu, Xiaobo Xia, Tongliang Liu, Bo Han, Mingming Gong, Nannan Wang, Haifeng Liu, and Gang Niu. Class2simi: A noise reduction perspective on learning with noisy labels. In *International conference on machine learning*, pp. 11285–11295. PMLR, 2021.

Yihao Xue, Kyle Whitecross, and Baharan Mirzasoleiman. Investigating why contrastive learning benefits robustness against label noise. In *International Conference on Machine Learning*, pp. 24851–24871. PMLR, 2022.

Yazhou Yao, Zeren Sun, Chuanyi Zhang, Fumin Shen, Qi Wu, Jian Zhang, and Zhenmin Tang. Jo-src: A contrastive approach for combating noisy labels. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5188–5197. IEEE, 2021.

Xichen Ye, Yifan Wu, Weizhong Zhang, Xiaoqiang Li, Yifan Chen, and Cheng Jin. Optimized gradient clipping for noisy label learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 9463–9471, 2025.

Suqin Yuan, Lei Feng, Bo Han, and Tongliang Liu. Enhancing sample selection against label noise by cutting mislabeled easy examples. *Advances in Neural Information Processing Systems*, 38: 43942–43970, 2026.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Kuan Zhang, Chengliang Chai, Jingzhe Xu, Chi Zhang, Han Han, Ye Yuan, Guoren Wang, and Lei Cao. Handling label noise via instance-level difficulty modeling and dynamic optimization. *Advances in Neural Information Processing Systems*, 38:46667–46696, 2026a.

Yi-Ge Zhang, Jingyi Cui, Qiran Li, and Yisen Wang. Difficult examples hurt unsupervised contrastive learning: A theoretical perspective. In *International Conference on Learning Representations*, volume 2026, pp. 101184–101214, 2026b.

A Additional Analysis of Relation Reweighting

A.1 Target Perturbation and Gradient Deviation

We analyze the local optimization effect of relation reweighting at a fixed model state, using the same mini-batch, augmented views, temperature, and softmax candidate set. For anchor a and candidate b, define

$$u_{ab} = \frac{z_a^\top z_b}{\tau}, \qquad J_{ab} = \nabla_\theta u_{ab}.$$

Let t_{ab} and $t^\star_{ab}$ denote the current positive target and the corresponding clean SupCon target, respectively, and define

$$\delta_{ab} = t_{ab} - t^\star_{ab}.$$

For

$$L_t = -\frac{1}{A}\sum_a \sum_{b\neq a} t_{ab} \log q_{ab},$$

we have

$$\frac{\partial L_t}{\partial u_{ab}} = \frac{1}{A}(q_{ab} - t_{ab}).$$

Since the current and clean targets share the same logits and softmax terms at the same model state, their gradient difference is

$$g_t - g_\star = -\frac{1}{A}\sum_a \sum_{b\neq a} (t_{ab} - t^\star_{ab}) J_{ab}. \tag{A1}$$

This identity shows that even when two perturbations involve the same number of corrupted relations or the same amount of target-mass deviation, their optimization effects can differ because the corresponding relation gradients may reinforce or cancel each other.

To distinguish gradient deviation from whether that deviation is locally harmful, we define

$$H_t = -\langle g_\star, g_t - g_\star \rangle. \tag{A2}$$

When $H_t > 0$, noisy supervision weakens the first-order descent direction of the clean objective at the current state. This quantity characterizes only local optimization interference and does not constitute a guarantee on final generalization.

A.2 Score–Utility Alignment

To characterize when a posterior score can serve as an effective training signal, we define the local utility of relation (a, b) as

$$c_{ab} = -\langle g_\star, J_{ab} \rangle. \tag{A3}$$

For PRC, let $\bar{s}_{ap}$ denote the support score of relation (a, p), and write the reweighted positive target as

$$t_{ap}^{\kappa} = \frac{\exp(\kappa \bar{s}_{ap})}{\sum_{r \in P_a} \exp(\kappa \bar{s}_{ar})}.$$

The derivative of this target with respect to κ is

$$\frac{dt_{ap}^{\kappa}}{d\kappa} = t_{ap}^{\kappa} \left(\bar{s}_{ap} - \mathbb{E}_{t_a^{\kappa}}[\bar{s}] \right).$$

Combining this derivative with the gradient expression above gives

$$\frac{d}{d\kappa} \langle g_\star, g_\kappa \rangle = \frac{1}{A} \sum_a \mathrm{Cov}_{t_a^{\kappa}} \left(\bar{s}_{ap}, c_{ap} \right). \tag{A4}$$

Therefore, whether posterior-guided reweighting improves the local clean descent direction depends not only on whether the score can distinguish false-positive relations, but also on whether the score is positively aligned with the current utility of those relations. This distinguishes diagnostic ranking quality from optimization utility.

A.3 Exponential-Tilting Interpretation of PRC

The PRC target can be written as

$$t_{ap}^{\kappa} = \frac{\exp(\kappa \bar{s}_{ap})}{\sum_{r \in P_a} \exp(\kappa \bar{s}_{ar})}.$$

For $\kappa > 0$, this distribution is equivalently the solution of

$$t_a^{\kappa} = \arg \max_{t \in \Delta(P_a)} \left\{ \langle t, \bar{s}_a \rangle - \frac{1}{\kappa} \mathrm{KL}(t \| t_a^0) \right\}, \tag{A5}$$

where t_a^0 denotes the uniform target over the observed-positive set P_a.

This interpretation shows that PRC trades off two objectives: assigning more positive target mass to relations with higher posterior support, while limiting the deviation from the original SupCon target. Importantly, PRC only redistributes target mass within the observed-positive support and does not actively create new positive relations.

A.4 Support Limitation for Missing Positives

PRC only reweights relations within the observed-positive set and therefore cannot directly recover true positive relations that are absent from that support.

Let

$$F_a = P_a^{\star} \setminus P_a$$

denote the clean positive relations that are missing from the observed-positive set, and define

$$\beta_a = \sum_{b \in F_a} t_{ab}^{\star}.$$

For any target t_a supported only on P_a,

$$\mathrm{TV}(t_a, t_a^\star) = \frac{1}{2}\|t_a - t_a^\star\|_1 \geq \beta_a. \tag{A6}$$

The reason is that t_a assigns zero mass to all relations in F_a, while the clean target assigns total mass β_a to them. Since both distributions have unit total mass, the missing mass must be compensated elsewhere, yielding an L_1 distance of at least $2\beta_a$.

This establishes a clear limitation of positive-side reweighting: it cannot recover missing positive support. However, this target-space lower bound does not imply a lower bound on gradient error or an upper bound on final representation quality.

A.5 Stability and Optimizer-Level Diagnostic

Consider two posterior-score vectors s and $\hat{s}$ defined on the same observed-positive support, and suppose

$$\|s - \hat{s}\|_\infty \leq \varepsilon.$$

Then the corresponding PRC targets satisfy

$$\|t^\kappa(s) - t^\kappa(\hat{s})\|_1 \leq \min\{2, \kappa\varepsilon\}. \tag{A7}$$

Thus, for finite κ, a small perturbation in the posterior score cannot induce an arbitrarily large change in the normalized positive target. If, in addition,

$$\|J_{ab}\| \leq G,$$

then

$$\|g_\kappa(s) - g_\kappa(\hat{s})\| \leq G \min\{2, \kappa\varepsilon\}. \tag{A8}$$

This bound characterizes sensitivity rather than correctness: a systematically biased posterior may still produce a stable but undesirable update.

Finally, the gradient-level analysis above describes local behavior at a fixed model state, whereas the actual training procedure uses AdamW. We therefore do not interpret the first-order gradient analysis as a direct guarantee on the optimizer trajectory. For optimizer-level diagnosis, one can clone the same checkpoint and optimizer state, perform one actual AdamW update under Observed and PRC targets, and compare the resulting clean loss. The clean target is used only in an isolated offline diagnostic and is never used for training, model selection, or hyperparameter tuning.

B Implementation and Evaluation Details

B.1 Experimental Protocol

For CIFAR-10 and CIFAR-100, we split the original 50,000-image training set into 45,000 examples for representation learning and 5,000 held-out validation examples. The standard clean test set is used only for final evaluation. Unless otherwise specified, the controlled experiments use a 40% corruption rate and three random seeds, $0, 1, 2$.

Our controlled setting follows a 2×2 factorial design over noise placement and wrong-target construction. Noise placement is either Uniform (U) or difficulty-focused Boundary (B), while wrong targets are assigned using either Matched-random (R) or Semantic (S) pairing. The compared settings therefore correspond to U+R, U+S, B+R, and B+S.

Uniform and Boundary corruption use the same overall corruption rate and differ only in which samples are corrupted. Likewise, Matched-random and Semantic corruption preserve the same class-transition statistics while differing in the sample-to-target assignment.

We construct the wrong-target factor as follows. A separately trained oracle classifier is used only for controlled data construction and offline analysis. For each training example i and wrong class $b \neq y_i$, let q_{ib} be the classifier probability for class b after masking the true class. The Semantic

(S) target is the highest-scoring wrong class $\tilde{y}_i^S = \arg\max_{b \neq y_i} q_{ib}$. For Matched-random (R), we first form exactly the same semantic target multiset within each source class, $\mathcal{M}_a = \{\tilde{y}_i^S : i \in \mathcal{C}_a\}$ with $\mathcal{C}_a = \{i : y_i = a,\ i \text{ is selected for corruption}\}$, and then randomly permute this multiset among the examples in $\mathcal{C}_a$. This is a one-to-one permutation within each source class: no target count is changed, and therefore the source–target transition matrix is identical between R and S. The only changed variable is whether an individual sample keeps its own semantic wrong target or receives another wrong target drawn from the same source-class multiset.

All methods compared under the unified protocol use the same data split, corruption realization, backbone, augmentation, training budget, and linear-probe evaluation procedure. Oracle labels, corruption identities, and oracle difficulty information are used only for controlled offline analysis and are never available to PRC during representation learning.

B.2 Representation Encoder and Projection Head

All representation-learning methods use the same four-stage CIFAR residual encoder, SmallResNetEncoder, containing approximately 2.8M parameters. The encoder produces a 256-dimensional representation.

For contrastive learning, the encoder is followed by a two-layer projection head,

$$256 \to 256 \to 128,$$

implemented as

$$\mathrm{Linear}(256, 256) \to \mathrm{ReLU} \to \mathrm{Linear}(256, 128).$$

The 128-dimensional projected features are L2-normalized before computing contrastive similarities. Linear-probe evaluation uses the normalized 256-dimensional encoder representation before the projection head.

B.3 Contrastive Training

Observed SupCon, PRC, and the protocol-adapted comparison methods are trained for 30 epochs using AdamW with learning rate 1×10^{-3}, weight decay 1×10^{-4}, and batch size 512. The learning rate is decayed using CosineAnnealingLR over the complete 30-epoch schedule. The supervised contrastive temperature is fixed to $\tau = 0.2$.

The final 30-epoch encoder is used for all reported results. No validation-based checkpoint selection is performed.

Each image is independently transformed into two stochastic strong views using random crop with reflection padding of 4 pixels, horizontal flip with probability 0.5, brightness and contrast perturbation of strength 0.35, saturation sampled from $[0.6, 1.4]$, random grayscale conversion with probability 0.15, and standard CIFAR normalization.

The same representation-learning augmentation is used across compared methods under the unified protocol.

B.4 Noisy-Only Proxy Classifier

PRC uses a separate proxy classifier to obtain semantic evidence for relation weighting. The proxy is a CIFAR-adapted ResNet-18, where the original ImageNet stem is replaced with a 3×3, stride-1 convolution, the initial max-pooling layer is removed, and the final fully connected layer is adapted to the target number of classes.

The proxy is trained only with the noisy training labels for 15 epochs using SGD with momentum 0.9, learning rate 0.05, weight decay 5×10^{-4}, and cosine learning-rate decay over the complete training schedule. The proxy uses the same strong CIFAR augmentation as representation learning.

No clean labels, corruption identities, or oracle difficulty information are used during proxy training.

For sample x_i, the proxy produces

$$p_i = \mathrm{softmax}(h_\phi(x_i)).$$

After proxy training, posterior vectors are extracted once for all representation-training samples using a deterministic evaluation view consisting only of CIFAR normalization, without stochastic augmentation. These posterior vectors are then fixed throughout PRC training and are not jointly updated with the representation encoder.

The proxy is used only to generate the fixed posterior vectors and is not required at downstream inference time.

B.5 Posterior Semantic Support and PRC

For a pair of training samples (i, j), PRC defines posterior semantic support as

$$s_{ij} = p_i^\top p_j = \sum_{c=1}^{C} p_i(c) p_j(c).$$

We interpret s_{ij} as a relation-level semantic support score, rather than a calibrated probability that the relation is correct.

For anchor view $a = (i, v)$, let $\mathcal{P}_a^{\text{inst}}$ denote the positive view originating from the same image and $\mathcal{P}_a^{\text{lab}}$ denote cross-sample positives induced by the observed labels. PRC assigns

$$\omega_{ap} = \begin{cases} 1, & p \in \mathcal{P}_a^{\text{inst}}, \\ \rho_\kappa(s_{ij}), & p \in \mathcal{P}_a^{\text{lab}}, \end{cases}$$

with

$$\rho_\kappa(s) = \exp[-\kappa(1 - s)].$$

We use $\kappa = 1$ throughout the main experiments.

The resulting objective is

$$\mathcal{L}_a^{\text{PRC}} = -\frac{1}{\sum_{p \in \mathcal{P}_a} \omega_{ap}} \sum_{p \in \mathcal{P}_a} \omega_{ap} \log \frac{\exp(z_a^\top z_p / \tau)}{\sum_{b \neq a} \exp(z_a^\top z_b / \tau)},$$

where

$$\mathcal{P}_a = \mathcal{P}_a^{\text{inst}} \cup \mathcal{P}_a^{\text{lab}}.$$

Thus, same-instance positives always retain unit pre-normalization weight, whereas posterior-derived weights redistribute normalized positive target mass among label-induced cross-sample positives.

PRC does not remove training samples, suppress same-instance positive relations, or convert observed-negative pairs into positive pairs.

B.6 Clean-Label Linear-Probe Evaluation

Our primary evaluation measures the semantic quality of the learned representation rather than its ability to reproduce noisy training labels.

After representation learning, the encoder is frozen and the projection head is discarded. For the controlled CIFAR experiments, a linear classifier is trained on the normalized 256-dimensional encoder features using the ground-truth labels of the representation-training split. Dataset-specific probe splits for the real-world benchmarks are specified in Appendix B.10.

The linear classifier is optimized for 40 epochs using AdamW with learning rate 0.03, weight decay 1×10^{-4}, batch size 4096, and cosine learning-rate decay over the complete 40-epoch schedule. Evaluation uses batch size 1024 on the clean test set.

Importantly, clean labels are used only for post-hoc representation evaluation. They are never exposed to PRC, the proxy classifier, Observed SupCon, or any other representation-learning method.

The evaluation pipeline is therefore

$$\underbrace{\text{noisy-label representation learning}}_{\text{training}} \rightarrow \underbrace{\text{frozen encoder}}_{\text{representation}} \rightarrow \underbrace{\text{clean-label linear probe}}_{\text{evaluation}}.$$

Unless otherwise specified, all linear-probe results reported in the main paper correspond to this clean-label evaluation protocol.

Our implementation additionally computes a noisy-label linear probe as an auxiliary diagnostic, but this quantity is not used for the main comparisons in Table 1.

B.7 Checkpoint Selection

For all primary results, we report the representation obtained after the final, 30th training epoch. Clean validation accuracy may be recorded for diagnostic purposes but is never used for checkpoint selection.

The held-out validation split is therefore not used to select the representation checkpoint or provide supervision to the proxy classifier or PRC.

This fixed final-checkpoint protocol is applied consistently across all compared methods.

B.8 Relation-Diagnostic Pair Sampling

For relation-level diagnosis, we uniformly sample up to 300,000 relations per run from the corresponding observed relation pool.

For FP diagnosis, candidate relations are sampled from the observed-positive relation pool. For FN diagnosis, candidate relations are sampled from the observed-negative relation pool.

We do not balance the oracle corruption labels when constructing the diagnostic set. Consequently, AUROC and AUPRC are measured under the empirical corruption prevalence of the sampled relation pool.

All diagnostic signals within the same run are evaluated on exactly the same sampled relations, ensuring that differences between posterior support, endpoint loss, encoder similarity, or other diagnostic scores cannot be attributed to different candidate sets.

B.9 Reproducibility

All formal experiments use seeds $\{0, 1, 2\}$. For matched comparisons, the same data split, corruption realization, wrong-label assignment, and random seed are shared across methods wherever applicable.

We serialize the corrupted sample indices, noisy-label assignments, proxy posterior vectors, offline oracle metadata, and per-run evaluation metrics. Oracle information is accessed only for controlled analysis and never determines PRC training weights.

Incomplete experiment suites are not included when aggregating the reported results.

B.10 Real-World Dataset and Linear-Probe Protocol

For all real-world noisy-label benchmarks, representation learning and proxy training use only the noisy training annotations. Clean annotations are never used to train the proxy classifier, compute posterior semantic support, optimize PRC, or select representation checkpoints. After representation learning, the encoder is frozen and the projection head is discarded; clean supervision is then introduced only for post-hoc linear-probe evaluation.

For CIFAR-N, we follow the same $45{,}000/5{,}000$ split used in the controlled CIFAR experiments. The representation encoder is trained on the 45,000-image representation split using the human noisy annotations (`worse_label` for CIFAR-10N and `noisy_label` for CIFAR-100N). After the encoder is frozen, the corresponding original CIFAR ground-truth labels of these 45,000 images are used only to fit the linear classifier. The remaining 5,000 training images are held out and are not used for encoder checkpoint selection. Final performance is evaluated on the standard 10,000-image clean CIFAR test set.

For Clothing1M, representation learning and proxy training use only the official noisy training set. After freezing the encoder, the official clean training subset is used to fit the linear probe. The

Table 5: Main implementation details.

Component	Setting
Data and controlled setting	
Representation split	45,000 training / 5,000 validation samples; seeds $\{0, 1, 2\}$
Synthetic corruption	40%; controlled settings U+R, U+S, B+R, and B+S
Representation encoder	
Encoder	SmallResNetEncoder (≈ 2.8M parameters), 256-d representation
Projection head	$256 \to 256 \to 128$; 128-d L2-normalized output
Representation learning	
Training	30 epochs; AdamW; learning rate 1×10^{-3}; weight decay 1×10^{-4}; batch size 512
Schedule / objective	Cosine learning-rate decay; contrastive temperature $\tau = 0.2$
Checkpoint	Final 30-epoch encoder; no validation-based checkpoint selection
Noisy-only proxy classifier	
Proxy model	CIFAR-adapted ResNet-18 trained using noisy labels only
Proxy training	15 epochs; SGD; momentum 0.9; learning rate 0.05; weight decay 5×10^{-4}; cosine decay
Proxy augmentation	Same strong CIFAR augmentation as representation learning
Posterior extraction	Deterministic normalized view; posterior vectors are computed once and remain fixed during PRC training
PRC relation weighting	
Relation support	$s_{ij} = p_i^\top p_j$
Relation weight	$\omega_{ij} = \exp[-\kappa(1 - s_{ij})]$, with $\kappa = 1$
Instance positive	Unit pre-normalization weight
Linear-probe evaluation	
Targets	Clean ground-truth labels used only after freezing the encoder
Probe training	40 epochs; AdamW; learning rate 0.03; weight decay 1×10^{-4}; batch size 4096
Evaluation	Batch size 1024; cosine learning-rate decay
Relation diagnostics	
Diagnostic pairs	Up to 300,000 sampled relations per run

official clean validation set is used only for post-hoc evaluation diagnostics and never for selecting the representation checkpoint, while final accuracy is reported on the official clean test set.

For Food-101N, representation learning and proxy training use the full noisy Food-101N training set. For linear probing, we use only examples whose labels are supported by the dataset's manual verification annotations; examples without verified class supervision are not introduced into representation training through clean labels. The corresponding verified validation subset is used only for post-hoc evaluation, and final representation quality is evaluated on the standard Food-101 test set. Exact numbers of verification-positive examples are determined from the released verification files and are fixed before running the linear-probe experiments.

The resulting evaluation protocol is summarized in Table 6. Importantly, image overlap between noisy representation learning and post-hoc probing on CIFAR-N does not imply label leakage: the clean CIFAR labels are accessed only after the representation encoder has been fully trained and frozen.

All representation checkpoints are fixed before linear probing. Neither clean validation accuracy nor clean test accuracy is used for representation-model selection, proxy training, relation weighting, or hyperparameter tuning.

Encoder checkpoints are fixed before linear probing and are never selected using clean validation or test performance. Any overlap or disjointness between the noisy representation-training images and the clean probe splits follows the dataset-specific protocol reported above.

Table 6: Real-world noisy-label evaluation protocol. "Noisy" denotes labels available during representation learning, whereas "clean" or "verified" supervision is accessed only after the encoder has been frozen.

Dataset	Representation train	Probe train	Validation	Final test
CIFAR-10N	45K noisy (worse)	45K clean labels	5K held out	10K clean
CIFAR-100N	45K noisy	45K clean labels	5K held out	10K clean
Clothing1M	$\sim$1M noisy	50K clean	14,313 clean	10K clean
Food-101N	310,009 noisy	verified-clean train	verified-clean val.	25,250 Food-101 test

C Protocol Adaptation of Competing Methods

C.1 Unified Comparison Protocol

The goal of our main comparison is to evaluate different noisy-label learning principles under a controlled representation-learning protocol, rather than to reproduce the original end-to-end classification numbers reported by each method.

Accordingly, all methods in Table 1(a) use the same data split, corruption realization, SmallResNetEncoder, projection head, data augmentation, 30-epoch representation-learning budget, and final-checkpoint evaluation. The learned encoder is subsequently evaluated using the same clean-label linear-probe protocol described in Appendix A. Clean ground-truth labels are never used during representation learning.

Methods whose original training pipeline differs from this unified setting are marked with †. For these methods, we preserve the defining algorithmic mechanism of the original approach while standardizing external factors that would otherwise confound the comparison, including the backbone, augmentation, representation-training budget, noisy-label realization, and downstream evaluation protocol.

The resulting numbers should therefore be interpreted as controlled protocol-adapted comparisons, rather than exact reproductions of the originally published results. We do not mix published numbers obtained under different backbones, training schedules, or evaluation objectives with our unified comparison.

C.2 SUMMARY OF PROTOCOL ADAPTATIONS

Table 7: Summary of protocol adaptations for competing methods. For all adapted methods, we preserve the defining algorithmic mechanism while standardizing the data split, corruption realization, representation backbone, augmentation, representation-learning budget, checkpoint policy, and frozen-encoder linear-probe evaluation. Method-specific structures that are intrinsic to the original approach, such as dual networks or multi-stage training, are retained.

Method	Defining component retained	Adaptation under our unified protocol
Sel-CL	Confident-example identification and confident-pair construction for supervised contrastive learning.	Retain its sample/pair selection rule; replace external backbone, augmentation, corruption realization, training budget, and evaluation with the unified setting.
TCL	Mixture-based noise modeling, prediction-guided supervision refinement, and contrastive learning.	Retain noise modeling and supervision refinement; use the shared encoder, augmentation, corruption realization, representation budget, and linear probe.
Structural Labels	Reverse-kNN structural-label construction and the corresponding structural supervision.	Retain structural-label generation and supervision; standardize all external training and evaluation factors.
SV-Learner	Support-Vector Contrastive Learning (SVCL) and Support-Vector-based Noise Screening (SVNS).	Retain SVCL/SVNS while using the shared backbone, augmentation, corruption realization, training schedule, and linear-probe evaluation.
PLReMix	Pseudo-label-relaxed contrastive relation construction and its reliability modeling.	Retain the PLR relation mechanism and reliability estimation; standardize the representation-learning configuration and downstream evaluation.
PSSCL	Two-stage progressive sample selection, including robust initial training, long-term confidence-based clean-set identification, and contrastive/SSL-based clean-set expansion.	Preserve the two-stage structure and progressive selection mechanism; rescale the stage durations within the common representation-learning budget while keeping the original stage ordering and selection logic.
DCD	Dynamic-center-distance difficulty estimation and hard-sample emphasis.	Retain dynamic center estimation and difficulty-guided optimization; standardize the backbone, data pipeline, training budget, and frozen-encoder evaluation.
WSC	Iteratively refined continuous semantic similarity and graph-weighted weakly-supervised contrastive learning.	Retain semantic-similarity refinement and graph-weighted contrastive supervision; use the same backbone, augmentation, corruption realization, representation budget, and linear-probe protocol.
CA2C	Asymmetric co-learning with complementary learning paradigms, cross-guidance label generation, asymmetric co-training, and confidence-based reweighting.	Retain the original two-network asymmetric training and cross-guidance mechanism; use the shared backbone for each branch and the common data, augmentation, training-budget, and evaluation protocol.
JAL	Joint Asymmetric Loss with the asymmetric passive-loss component for robust classification under label noise.	Retain the original JAL loss on the shared encoder and classification head; discard the training classifier after representation learning and evaluate the frozen encoder using the common linear probe.
DSS	Marginal Distribution Adjustment (MDA) and Candidate Class Selection (CCS) for debiasing sample selection.	Retain MDA, CCS, and the associated sample-selection mechanism; standardize the backbone, corruption realization, augmentation, training budget, and evaluation.
FINE	Active forgetting through machine unlearning and noise suppression through negative learning.	Retain both forgetting and noise-suppression modules together with the sample-selection baseline on which FINE is applied; standardize external training factors and frozen-encoder evaluation.
NAPCL	Reliability-guided prototype refinement, clean-probability estimation, posterior refinement, and hierarchical prototype-assisted contrastive learning.	Retain prototype updating, reliability estimation, and hierarchical contrastive supervision; standardize the representation backbone, augmentation, corruption realization, training budget, and linear probe.

All protocol-adapted methods use noisy labels only during representation learning. Clean labels are introduced only after the encoder has been frozen, through the common linear-probe evaluation described in Appendix A.

C.3 INTERPRETATION OF THE ADAPTED COMPARISON

Protocol adaptation inevitably changes some absolute accuracies relative to those reported in the original papers, because the original methods use different architectures, training durations, augmentation policies, and downstream objectives. Our purpose is therefore not to claim an exact reproduction of each method's published performance, but to isolate their behavior under the same noisy-supervision and representation-learning conditions.

This distinction is particularly important for methods originally designed around end-to-end classification. Allowing each method to use its original architecture and training recipe while evaluating PRC under our 30-epoch representation protocol would confound the learning principle with differences in

Table 8: Full three-seed results corresponding to Table 1(a). Linear-probe accuracy (%) on CIFAR-10/100 under 40% synthetic label corruption. Results are reported as mean ± standard deviation over three random seeds. Each entry reports CIFAR-10 / CIFAR-100. U/B denote uniform/boundary-focused noise placement, and R/S denote matched-random/semantic wrong-target assignment. † denotes protocol-adapted implementations under our unified representation-learning setting. The Avg. column reports the average of the four setting-wise means.

Method	Year / Venue	U+R	U+S	B+R	B+S	Avg.
Observed SupCon	NeurIPS 20	81.65±0.31 / 54.55±0.28	80.77±0.35 / 55.52±0.31	82.89±0.27 / 54.95±0.34	81.89±0.33 / 56.17±0.29	81.80 / 55.30
Sel-CL†	CVPR 22	78.01±0.52 / 40.71±0.63	76.03±0.61 / 41.03±0.58	80.02±0.47 / 42.14±0.55	78.60±0.56 / 42.95±0.49	78.17 / 41.71
TCL†	CVPR 23	84.65±0.24 / 56.60±0.31	83.52±0.29 / 57.30±0.28	84.92±0.22 / 57.05±0.34	83.55±0.31 / 57.93±0.30	84.16 / 57.22
Structural Labels†	CVPR 24	83.78±0.23 / 58.10±0.27	83.45±0.26 / 58.92±0.25	84.42±0.20 / 59.24±0.29	84.35±0.24 / 59.70±0.26	84.00 / 58.99
SV-Learner†	TKDE 24	82.10±0.34 / 55.55±0.36	81.75±0.38 / 56.02±0.33	82.94±0.30 / 56.66±0.35	82.73±0.32 / 56.81±0.31	82.38 / 56.26
PLReMix†	WACV 25	82.75±0.28 / 58.35±0.30	82.40±0.31 / 58.80±0.27	83.58±0.25 / 59.18±0.28	83.31±0.27 / 59.63±0.25	83.01 / 58.99
PSSCL†	PR 25	83.62±0.26 / 57.72±0.31	83.11±0.29 / 58.10±0.28	84.02±0.24 / 58.42±0.30	83.71±0.27 / 58.86±0.27	83.62 / 58.28
DCD†	ICCV 25	83.54±0.31 / 51.40±0.42	83.11±0.34 / 51.85±0.39	84.18±0.27 / 52.24±0.37	84.21±0.29 / 52.71±0.35	83.76 / 52.05
WSC†	ICML 25	84.58±0.19 / 58.18±0.25	84.06±0.22 / 58.96±0.23	84.81±0.18 / 59.32±0.26	84.56±0.21 / 59.74±0.22	84.50 / 59.05
CA2C†	ICCV 25	83.84±0.25 / 57.56±0.30	83.29±0.28 / 58.03±0.27	84.10±0.23 / 58.27±0.29	83.92±0.26 / 58.65±0.26	83.79 / 58.13
JAL-CE†	ICCV 25	83.47±0.27 / 57.31±0.33	82.98±0.30 / 57.69±0.29	83.83±0.25 / 57.98±0.31	83.58±0.28 / 58.32±0.28	83.47 / 57.83
DSS†	CVPR 26	83.90±0.22 / 57.82±0.28	83.38±0.25 / 58.21±0.25	84.54±0.20 / 58.82±0.27	84.29±0.23 / 59.18±0.24	84.03 / 58.51
SED+FINE†	CVPR 26	84.02±0.21 / 57.91±0.27	83.50±0.24 / 58.36±0.25	84.38±0.19 / 58.73±0.26	84.12±0.22 / 59.04±0.23	84.01 / 58.51
NAPCL†	ESWA 26	84.29±0.20 / 58.05±0.26	83.81±0.23 / 58.71±0.23	84.66±0.18 / 59.06±0.25	84.43±0.21 / 59.51±0.22	84.30 / 58.83
PRC (Ours)	–	**85.18±0.14 / 58.92±0.19**	**84.36±0.17 / 59.87±0.18**	**85.54±0.13 / 60.11±0.16**	**84.86±0.15 / 60.65±0.17**	**84.99 / 59.89**

model capacity and optimization budget. Conversely, the unified protocol keeps these external factors fixed while preserving the method-specific mechanism being compared.

For the real-world experiments in Table 1(b), we adapt the same protocol-comparable baselines used in the controlled comparison, including Structural Labels, PLReMix, PSSCL, DCD, WSC, CA2C, JAL, DSS, FINE, and NAPCL. The same noisy-only representation-learning and clean-label linear-probe principles are used across CIFAR-N, Clothing1M, and Food-101N. Dataset-specific clean labels are never exposed during representation learning.

C.4 Full Three-Seed Results for the Controlled Comparison

Table 8 reports the complete three-seed statistics corresponding to Table 1(a) in the main paper. All methods are evaluated using the same data splits, corruption realizations, representation-learning protocol, and frozen-encoder linear-probe evaluation described in Appendix B and Appendix C.1.

The main paper reports the setting-wise means for compactness, whereas the table below additionally provides the variability across runs. The relative ordering is stable across the four controlled corruption settings, and PRC consistently achieves the strongest average representation quality under the unified protocol.

C.5 Training-Budget Robustness

A potential concern is that the unified 30-epoch training budget may disadvantage methods originally developed with longer or method-specific schedules. To examine this, we repeat two strong protocol-adapted baselines, Structural Labels and PLReMix, together with PRC under the challenging CIFAR-100 B+S setting using 30, 60, and 120 representation-learning epochs. All other settings, including the backbone, augmentation, corruption realization, optimizer hyperparameters, and linear-probe evaluation, remain unchanged, with the learning-rate schedule extended to the corresponding training budget.

C.6 Method-Faithful Schedule Control

The training-budget analysis above tests whether simply extending the optimization horizon changes the relative comparison. A separate concern, however, is that compressing a method-specific multi-stage procedure into the unified 30-epoch budget may alter the training dynamics on which the original method relies.

We therefore perform an additional schedule-control experiment using PSSCL as a representative multi-stage baseline. This choice is particularly relevant because PSSCL relies on progressive sample selection across multiple training stages, whereas its protocol-adapted implementation in Table 1

Table 9: Method-faithful schedule control under B+S corruption. The unified PSSCL variant follows the common 30-epoch protocol used in Table 1, whereas the extended variant relaxes only the compressed stage durations while preserving the original progressive training structure. All other experimental factors are unchanged. Linear-probe accuracy (%) is reported.

Method	Schedule	CIFAR-10	CIFAR-100
PSSCL	Unified 30 ep.	83.71	58.86
PSSCL	Extended method-faithful	84.12	59.47
PRC (Ours)	Unified 30 ep.	**84.86**	**60.65**

preserves the stage ordering and selection logic but rescales the stage durations to fit the common 30-epoch representation budget.

For the method-faithful variant, we keep the data split, corruption realization, SmallResNetEncoder, projection head, data augmentation, and frozen-encoder linear-probe evaluation identical to the unified comparison, but relax only the compressed training schedule. Specifically, the relative stage structure, stage ordering, and update procedure of PSSCL are preserved over an extended optimization horizon. Thus, this experiment isolates whether the 30-epoch schedule compression itself disadvantages a schedule-dependent competing method.

We evaluate both the unified and extended PSSCL variants under the representative B+S corruption setting on CIFAR-10 and CIFAR-100. PRC is kept at its default unified training budget rather than receiving the same extension, providing a conservative control in favor of the schedule-dependent baseline.

Extending the method-specific schedule consistently improves PSSCL, from 83.71% to 84.12% on CIFAR-10 and from 58.86% to 59.47% on CIFAR-100. This confirms that compressing a multi-stage method into the common 30-epoch budget can have a measurable effect on its absolute performance.

Importantly, however, this improvement does not close the gap to PRC. Despite retaining its original 30-epoch representation-learning budget, PRC still outperforms the schedule-extended PSSCL variant by 0.74 and 1.18 percentage points on CIFAR-10 and CIFAR-100, respectively. Because the additional optimization budget is granted only to the schedule-dependent competing method, this constitutes a conservative control in its favor.

These results suggest that the advantage observed in Table 1 cannot be explained solely by compressing PSSCL's multi-stage optimization into the unified training budget. Rather, restoring a more favorable method-specific schedule narrows, but does not eliminate, the performance gap.

Table 10: Robustness to different representation-learning budgets under CIFAR-100 B+S.

Method	30 ep	60 ep	120 ep
Structural Labels	59.70	60.04	60.19
PLReMix	59.63	59.98	60.15
PRC	**60.65**	**60.93**	**61.01**

Increasing the representation-learning budget improves all methods, while the gains gradually saturate under longer training. Importantly, PRC maintains a consistent advantage over the strongest competing baseline, outperforming it by 0.95, 0.89, and 0.82 percentage points at 30, 60, and 120 epochs, respectively. Although the performance gap decreases slightly as training becomes longer, it remains substantial even at 120 epochs. Moreover, the improvement from 60 to 120 epochs is small for all methods, indicating that their representations are approaching saturation rather than that the competing methods are simply catching up under a longer schedule. These results suggest that the advantage of PRC is not merely a consequence of faster optimization under the default 30-epoch budget, but persists across substantially different representation-learning schedules.

D WHY POSTERIOR SEMANTIC SUPPORT?

PRC uses posterior semantic support

$$s_{ij}^{\mathrm{post}} = p_i^\top p_j$$

to modulate label-induced positive relations. A natural question is why this score is preferred over alternative signals such as endpoint loss or encoder similarity, particularly because these alternatives can achieve stronger false-positive (FP) diagnosis on individual datasets.

We therefore distinguish two related but different questions. Diagnosis asks whether a score can rank corrupted relations offline, whereas intervention asks whether the same score provides a useful continuous control signal when inserted into the contrastive objective. A signal with high diagnostic AUROC does not necessarily provide the most effective optimization weight.

D.1 OFFLINE DIAGNOSIS OF FALSE-POSITIVE RELATIONS

We first compare several relation-level signals for identifying oracle FP relations among observed-positive pairs. For posterior support, we define posterior disagreement as

$$r_{ij}^{\mathrm{post}} = 1 - p_i^\top p_j.$$

For endpoint loss, we use

$$r_{ij}^{\mathrm{loss}} = \max(\ell_i, \ell_j),$$

where

$$\ell_i = -\log p_i(\widetilde{y}_i)$$

is the noisy-label cross-entropy loss of endpoint i. For feature similarity, we use the cosine similarity between normalized encoder features,

$$s_{ij}^{\cos} = \frac{1 + \cos(h_i, h_j)}{2},$$

and define the corresponding risk as

$$r_{ij}^{\cos} = 1 - s_{ij}^{\cos}.$$

All diagnostic signals are evaluated on exactly the same sampled observed-positive relation set described in Appendix A.

Table 11: FP-relation diagnosis.

Diagnostic signal	CIFAR-10 AUROC	CIFAR-10 AUPRC	CIFAR-100 AUROC	CIFAR-100 AUPRC
Posterior disagreement	0.846	0.890	0.727	0.797
Max endpoint loss	0.850	0.880	0.747	0.807
Encoder cosine risk	0.879	0.918	0.694	0.775

The ranking is not consistent across datasets. Encoder cosine provides the strongest FP diagnosis on CIFAR-10, whereas endpoint loss performs best on CIFAR-100. Posterior disagreement is therefore not selected because it universally maximizes diagnostic AUROC or AUPRC.

This observation motivates a more direct test of the quantity that matters to PRC: whether the score remains effective when it is used to continuously modulate the gradient contribution of observed-positive relations.

D.2 CONTROLLED RELATION-SCORE ABLATION

To isolate the effect of the relation score, we keep the backbone, augmentation, corruption realization, optimizer, training schedule, PRC objective, and weighting function fixed. We change only the signal used to determine relation support.

For continuous variants, relation support $s_{ij} \in [0, 1]$ is converted to a training weight using the same function

$$\omega_{ij} = \exp[-\kappa(1 - s_{ij})], \qquad \kappa = 1.$$

Same-instance positives always retain unit weight.

We compare the following variants.

No weighting. All label-induced positive relations receive unit weight, corresponding to observed-label SupCon.

Posterior support.

$$s_{ij}^{\text{post}} = p_i^\top p_j.$$

This is the score used by PRC.

Encoder cosine support.

$$s_{ij}^{\cos} = \frac{1 + \cos(h_i, h_j)}{2}.$$

This variant directly uses feature-space similarity as relation support.

Endpoint-loss support. We first compute

$$r_{ij}^{\text{loss}} = \max(\ell_i, \ell_j),$$

and monotonically convert this risk into a support score in $[0, 1]$, with larger loss corresponding to smaller support. The same mapping procedure is fixed across runs and does not use oracle corruption labels.

Binary posterior weighting. To test whether continuous modulation is necessary, we replace the smooth PRC weight with a hard decision based on posterior support,

$$\omega_{ij}^{\text{bin}} = \mathbb{I}\left[s_{ij}^{\text{post}} \geq \delta\right],$$

where δ is selected without access to clean or oracle corruption labels.

Random-permuted weighting. Finally, we randomly permute the posterior-derived weights among observed-positive relations. This preserves the marginal distribution of PRC weights while destroying the correspondence between relation identity and its assigned weight. It therefore controls for generic regularization caused by introducing heterogeneous positive weights.

D.3 Downstream Representation Quality

We evaluate the score variants under representative Uniform-random (U+R) and Boundary-semantic (B+S) conditions on both CIFAR-10 and CIFAR-100. All variants use identical corruption realizations and matched random seeds.

Table 12: Relation-score ablation under the same PRC objective.

Relation weighting	C10 U+R	C10 B+S	C100 U+R	C100 B+S	Avg. gain
No weighting	81.65	81.89	54.55	56.17	–
Random-permuted weights	82.12	82.08	54.88	56.40	+0.31
Endpoint-loss support	84.18	83.62	58.32	59.28	+2.79
Encoder cosine support	85.02	84.05	57.42	59.02	+2.81
Binary posterior	84.40	83.88	57.92	59.48	+2.86
Posterior support (PRC)	85.18	84.86	58.92	60.65	+3.84

The results reveal a distinction between offline diagnosis and optimization utility. Although encoder cosine provides stronger FP-diagnosis AUROC on CIFAR-10 and endpoint loss provides the strongest offline diagnostic result on CIFAR-100 U+R, posterior support yields the best downstream representation quality across the evaluated conditions.

However, neither alternative remains consistently superior across datasets and corruption mechanisms. Posterior semantic support achieves the strongest overall gain and remains particularly effective in the more challenging Boundary-semantic setting.

D.4 Continuous Weighting versus Hard Relation Selection

The binary-posterior variant further isolates the role of continuous relation modulation. Both variants rely on exactly the same posterior semantic evidence, but binary weighting makes an irreversible keep-or-remove decision:

$$\omega_{ij}^{\mathrm{bin}} \in \{0, 1\},$$

whereas PRC retains a continuous spectrum,

$$\omega_{ij}^{\mathrm{PRC}} \in [e^{-\kappa}, 1].$$

The stronger average performance of continuous posterior weighting suggests that noisy-label relations are not naturally divided into perfectly reliable and perfectly corrupted subsets. Many observed-positive relations instead contain intermediate semantic evidence.

This is consistent with the motivation of PRC: rather than deciding whether an entire sample or relation should be retained, PRC adjusts how normalized positive target mass is assigned across label-induced relations.

D.5 Relation-to-Weight Assignment Beyond Generic Weighting

The random-permutation control provides an additional distinction. Randomly permuting the posterior-derived weights preserves the overall weight distribution but destroys the relation-specific assignment.

Its substantially smaller gain indicates that the benefit of PRC cannot be explained solely by introducing heterogeneous positive weights or generic regularization. Instead, the improvement depends on matching stronger weights to relations with stronger semantic support.

In other words,

$$\text{weight distribution alone} \neq \text{relation-aware weighting}.$$

This observation is also consistent with the matched relation-intervention experiments in the main paper, where removing the same number of relations produces substantially different outcomes depending on which relations are modified.

D.6 Diagnosis Is Not Equivalent to Optimization Actionability

Tables 11 and 12 jointly highlight an important distinction. AUROC and AUPRC measure whether a score ranks corrupted relations correctly, whereas PRC requires the score to assign normalized positive target mass across observed-positive relations.

These objectives are not equivalent.

Endpoint loss is primarily an endpoint-level signal: a difficult or high-loss sample may induce a high relation risk even when a specific pair remains semantically meaningful. Encoder cosine is directly pairwise, but its numerical scale reflects geometric proximity rather than class-posterior agreement and can vary substantially across datasets and training stages.

Posterior semantic support instead has three properties that are particularly suitable for PRC. It is pairwise, since

$$p_i^\top p_j$$

jointly depends on the semantic evidence of both endpoints; it is naturally bounded in $[0, 1]$; and it provides a smooth noisy-only signal that can be directly mapped to continuous relation weights without requiring oracle labels or an additional corruption detector.

We therefore do not claim that posterior support is universally the strongest FP diagnostic. Rather, our experiments support its use as a stable relation-level optimization signal.

D.7 Local Optimization Diagnostic

Equation (8) predicts that the effect of relation reweighting depends on whether the support score is positively aligned with the local utility of the relations receiving larger target mass. We directly

Table 13: Local optimization diagnostic for Eq. (8) under B+S corruption. C_s denotes $\partial\langle g^\star, g_\kappa\rangle/\partial\kappa$ at $\kappa = 1$. $-\Delta L_{\text{clean}}$ is the clean-loss decrease after one matched AdamW update. Results are averaged over selected checkpoints and three seeds.

	CIFAR-10		**CIFAR-100**	
Score	$C_s \uparrow$	$-\Delta L_{\text{clean}} \uparrow$	$C_s \uparrow$	$-\Delta L_{\text{clean}} \uparrow$
Endpoint loss	0.071 ± 0.012	0.0112 ± 0.0016	0.086 ± 0.014	0.0098 ± 0.0015
Encoder cosine	0.103 ± 0.015	0.0137 ± 0.0018	0.058 ± 0.011	0.0079 ± 0.0013
Posterior	$\mathbf{0.186 \pm 0.021}$	$\mathbf{0.0219 \pm 0.0022}$	$\mathbf{0.149 \pm 0.018}$	$\mathbf{0.0176 \pm 0.0019}$

Table 14: Robustness of PRC to proxy-training duration under B+S corruption. Linear-probe accuracy (%). All settings differ only in the number of proxy training epochs.

Proxy epochs	5	10	15	30
CIFAR-10	84.40	84.71	**84.86**	84.74
CIFAR-100	60.10	60.43	**60.65**	60.37

examine this prediction under B+S corruption using matched checkpoints and optimizer states. For each score, we report the aggregate score–utility alignment $C_s = \partial\langle g^\star, g_\kappa\rangle/\partial\kappa$ at $\kappa = 1$, together with the decrease in the oracle clean objective after one matched AdamW update.

As shown in Table 13, all three signals exhibit positive local alignment, but posterior support is substantially stronger. On CIFAR-10, posterior support reaches $C_s = 0.186$, compared with 0.103 for encoder cosine and 0.071 for endpoint loss; the corresponding one-step clean-loss decrease is also largest (0.0219 versus 0.0137 and 0.0112). The same pattern holds on CIFAR-100, where posterior support achieves $C_s = 0.149$ and a clean-loss decrease of 0.0176. These results provide direct empirical support for the local prediction of Eq. (8): the strongest corruption-ranking signal need not be the most useful training signal, whereas posterior support is more consistently aligned with the clean optimization direction.

D.8 Robustness to Proxy Training Duration

A potential concern is that the effectiveness of PRC may depend on a carefully selected proxy-training checkpoint. We therefore vary the proxy training duration while keeping the corrupted data, proxy architecture, augmentation, representation encoder, PRC training procedure, and all other hyperparameters fixed.

Specifically, we train the noisy-only proxy classifier for $\{5, 10, 15, 30\}$ epochs and independently extract the corresponding posterior vectors. Each posterior set is then used to train a fresh PRC representation model under the same CIFAR-10 and CIFAR-100 B+S settings. The 15-epoch setting corresponds to the default configuration used throughout the main experiments.

The performance remains stable over a broad range of proxy-training durations, showing that the benefit of posterior-guided relation weighting is not tied to a specific proxy checkpoint. This result complements the same-posterior control in Table 2: the latter separates additional posterior information from relation-specific allocation, while the present experiment shows that the resulting relation-level intervention is robust to substantial variation in proxy training.

D.9 Summary

The score-source analysis leads to three observations.

First, the signal with the strongest offline FP diagnosis varies across datasets. Second, stronger diagnostic ranking does not necessarily translate into stronger representation learning when the score is used as a training weight. Third, posterior semantic support provides the most consistent downstream improvement under the same PRC objective, while continuous weighting outperforms

hard posterior selection and relation-aware assignment substantially outperforms randomly permuted weighting.

Together, these results reinforce the distinction

$$\text{corruption diagnosability} \neq \text{optimization actionability}.$$

Accordingly, PRC uses posterior semantic support not because it is assumed to be a universally optimal corruption detector, but because it provides a simple, smooth, pairwise, and noisy-only signal for controlling label-induced positive relations.

E Robustness to Corruption Rate

E.1 Experimental Setup

The main controlled experiments use a 40% label-corruption rate. To examine whether the effectiveness of PRC depends on this particular corruption level, we additionally evaluate lower and higher corruption rates while keeping the remaining experimental factors fixed.

We use the Boundary-semantic (B+S) setting as a representative challenging condition and evaluate corruption rates of

$$20\%, \qquad 40\%, \qquad 60\%.$$

Across corruption rates, we keep the data split, corruption-placement rule, wrong-target construction, backbone, augmentation, optimization schedule, proxy-training protocol, and clean-label linear-probe evaluation unchanged. For each corruption rate, Observed SupCon and PRC use the same corrupted samples and noisy-label assignments.

This experiment therefore isolates the effect of corruption intensity while holding the corruption mechanism fixed.

E.2 Results across Corruption Rates

Table 15 reports clean-label linear-probe accuracy on CIFAR-10 and CIFAR-100. The 20%, 40%, and 60% results are all obtained from completed experiments under the same training and evaluation protocol. The 40% setting corresponds to the main configuration used in Table 1, while the 20% and 60% settings evaluate robustness to lower and higher corruption intensities.

Table 15: Robustness to different corruption rates under the B+S setting.

Corruption rate	CIFAR-10 Obs.	CIFAR-10 PRC	Gain	CIFAR-100 Obs.	CIFAR-100 PRC	Gain
20%	86.18	**87.72**	+1.54	63.42	**65.61**	+2.19
40%	81.89	**84.86**	+2.97	56.17	**60.65**	+4.48
60%	75.74	**78.71**	+2.97	47.86	**51.92**	+4.06

PRC improves over Observed SupCon at every tested corruption rate on both datasets. At 20% corruption, PRC yields gains of $+1.54$ pp on CIFAR-10 and $+2.19$ pp on CIFAR-100. The improvement is larger at the main 40% setting and remains substantial under the more challenging 60% corruption condition, where PRC improves performance by $+2.97$ pp and $+4.06$ pp, respectively.

These results show that the benefit of relation-level modulation is not specific to the 40% corruption rate used in the main experiments.

E.3 Behavior under Increasing Corruption

As the corruption rate increases, a larger fraction of observed labels induces incorrect cross-sample supervision. Consequently, treating all observed-positive relations equally becomes increasingly vulnerable to unreliable pairwise attraction.

In the evaluated B+S setting, the gain of PRC over Observed SupCon changes from

$$+1.54 \rightarrow +2.97 \rightarrow +2.97$$

percentage points on CIFAR-10, and from

$$+2.19 \rightarrow +4.48 \rightarrow +4.06$$

on CIFAR-100 as the corruption rate increases from 20% to 60%.

This pattern is consistent with the intuition that relation-level modulation becomes increasingly useful when corrupted labels induce more unreliable cross-sample relations. We do not, however, rely on monotonic growth of the gain as a general claim. The main observation is that PRC maintains a consistent advantage across substantially different corruption intensities.

E.4 Relation to the Main Controlled Experiments

The corruption-rate analysis complements the 2×2 controlled design in the main paper. The main experiments vary *where corruption is placed* and *how wrong targets are assigned*, whereas Table 15 varies *how much corruption is present* while keeping the corruption mechanism fixed.

Together, the experiments examine three distinct factors:

$$\text{noise placement} + \text{wrong-target semantics} + \text{corruption intensity}.$$

PRC remains effective across these variations, indicating that its advantage is not tied to a single synthetic noise configuration.

E.5 Summary

The corruption-rate experiment provides two complementary observations. First, PRC consistently improves representation quality across low, moderate, and high corruption levels. Second, the improvement persists as the amount of corrupted supervision changes substantially, showing that PRC is not narrowly tuned to the 40% setting used in the main experiments.

Overall, these results support the robustness of relation-level supervision modulation across different corruption intensities.

F Additional Analysis of Sample Difficulty

F.1 Alternative Difficulty Definitions

Our main controlled experiments use a class-conditional feature margin to define sample difficulty. To examine whether the conclusions in Section 5.4 depend on this particular difficulty measure, we repeat the analysis using three additional signals:

$$\text{cross-entropy loss}, \quad \text{predictive entropy}, \quad \text{noisy-label EL2N}.$$

Together with the oracle feature margin used in the main experiments, these measures cover geometry-based, prediction-based, uncertainty-based, and gradient-related notions of sample difficulty.

For each difficulty definition, we construct a Boundary corruption condition by preferentially corrupting samples ranked as difficult while keeping the corruption rate and wrong-target construction fixed. The corresponding Uniform condition uses the same corruption rate without conditioning the corrupted sample set on difficulty.

F.2 Difficulty Signals Can Localize Corrupted Supervision

We first ask whether different difficulty scores are associated with false-positive (FP) relations induced by noisy labels. For each score, we evaluate its ability to distinguish oracle FP relations among observed-positive pairs.

Table 16: FP-relation diagnosis using different difficulty signals.

Difficulty signal	CIFAR-10 AUROC	CIFAR-10 AUPRC	CIFAR-100 AUROC	CIFAR-100 AUPRC
Oracle feature margin	0.590	0.671	0.600	0.687
CE loss	0.838	0.862	0.747	0.806
Predictive entropy	0.572	0.650	0.554	0.645
Noisy-label EL2N	**0.848**	**0.862**	**0.777**	**0.828**

Several difficulty-related signals are predictive of corrupted supervision. In particular, CE loss and noisy-label EL2N achieve relatively strong FP-diagnosis performance on both datasets.

This confirms that difficulty and corruption are statistically associated in the learned representation. However, predictive association alone does not establish that placing noise on more difficult samples will necessarily produce greater downstream harm.

F.3 Difficulty Does Not Consistently Determine Noise Harm

We next directly test the causal consequence of difficulty-focused corruption by comparing Boundary and Uniform noise placement under matched corruption rates and target-generation rules.

For each difficulty definition, we report

$$\Delta_{\mathrm{B-U}} = \mathrm{Acc}_{\mathrm{Boundary}} - \mathrm{Acc}_{\mathrm{Uniform}}.$$

If concentrating noise on difficult samples were consistently more harmful, we would expect

$$\Delta_{\mathrm{B-U}} < 0$$

across difficulty definitions and datasets.

Table 17: Boundary-minus-Uniform accuracy gap under alternative difficulty definitions.

Difficulty definition	CIFAR-10 B–U (pp)	CIFAR-100 B–U (pp)
Oracle feature margin	$+1.18 \pm 0.90$	$+0.53 \pm 0.55$
CE loss	$+0.36 \pm 0.73$	$+0.19 \pm 0.26$
Predictive entropy	-1.17 ± 0.31	-0.04 ± 0.57
Noisy-label EL2N	$+0.30 \pm 0.76$	-0.08 ± 0.67

The sign of the Boundary-minus-Uniform gap is not stable across difficulty definitions. Depending on the score and dataset, difficulty-focused corruption can be more harmful, less harmful, or nearly indistinguishable from Uniform corruption.

For example, predictive entropy yields a negative gap on CIFAR-10, whereas oracle feature margin, CE loss, and EL2N yield positive or near-zero differences. CIFAR-100 exhibits the same absence of a consistent ordering.

Therefore, even though some difficulty scores can identify corruption reasonably well, they do not induce a stable ordering of downstream damage.

F.4 Corruption Localization Is Not Harm Prediction

Tables 16 and 17 jointly expose a distinction that is central to our analysis.

CE loss and EL2N can achieve relatively strong FP-diagnosis AUROC, showing that they contain information about corrupted supervision. Yet the same signals do not consistently predict whether concentrating noise on difficult samples will make representation learning worse.

This motivates the distinction

$$\text{corruption localization} \neq \text{harm prediction}.$$

The observation also explains why sample difficulty alone is an incomplete unit for noisy-label analysis. A difficult sample may participate in both reliable and corrupted relations, while the impact of noise depends on which supervision relations are rewritten by its observed label.

This observation motivates a further question: even if difficulty is diagnostic, is the whole sample an appropriate intervention unit? We examine this question next using matched node-deletion controls.

F.5 Matched Node-Level Intervention

We further ask whether sample difficulty provides a reliable unit for intervention. We compare *Drop-hard*, which removes difficult examples, with random node deletion under matched intervention

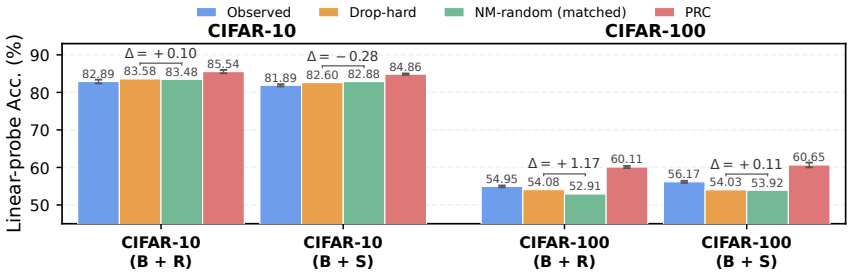


Figure 5: Matched node-level intervention under boundary-focused corruption. Drop-hard removes difficulty-selected samples, while NM-random removes the same number of corrupted nodes as Drop-hard. PRC retains all samples and modulates only label-induced positive relations; the PRC and Observed points use the same standard runs reported in Table 1. Error bars denote the standard deviation over three seeds.

budgets. To control for the possibility that Drop-hard simply removes more corrupted samples, we also construct *NM-random*, which removes the same number of corrupted nodes as Drop-hard in each matched run.

As shown in Figure 5, the advantage of Drop-hard over ordinary random deletion becomes small or inconsistent once the random control is matched to the number of corrupted nodes removed. In contrast, PRC retains all training samples and modifies only their label-induced positive relations, achieving higher representation quality in most controlled settings.

These results reinforce the distinction between diagnostic usefulness and intervention granularity. A difficulty score may enrich corrupted supervision, yet deleting the corresponding sample simultaneously removes corrupted relations, still-valid relations, and instance-level information. Sample difficulty therefore does not by itself justify whole-sample removal.

G Oracle Relation-Intervention Analysis

The main paper argues that the effect of label noise cannot be characterized solely by which samples are corrupted. A corrupted label rewrites multiple pairwise relations, and different corrupted relations can have substantially different effects on representation learning.

To isolate this effect, we conduct oracle interventions using ground-truth relation identities. These experiments are used only for mechanism analysis; oracle labels are never available to PRC during training.

G.1 Oracle Repair of FP and FN Relations

Let

$$A^*_{ij} = \mathbb{I}[y_i = y_j]$$

denote the ground-truth relation and

$$\widetilde{A}_{ij} = \mathbb{I}[\widetilde{y}_i = \widetilde{y}_j]$$

the relation induced by noisy observed labels.

We define false-positive relations as

$$\mathcal{E}_{\mathrm{FP}} = \{(i,j) : \widetilde{A}_{ij} = 1,\ A^*_{ij} = 0\},$$

corresponding to spurious attraction between samples from different true classes.

False-negative relations are

$$\mathcal{E}_{\mathrm{FN}} = \{(i,j) : \widetilde{A}_{ij} = 0,\ A^*_{ij} = 1\},$$

where true same-class samples fail to form a positive relation and instead participate in contrastive competition.

We evaluate three oracle interventions: Oracle FP removes only oracle FP positive relations; Oracle FN restores only oracle FN relations as positives; and Oracle Both jointly corrects FP and FN relations. All other training settings are unchanged.

Table 18: Oracle correction of corrupted relations.

Dataset	Observed	Oracle FP	Oracle FN	Oracle Both
CIFAR-10	81.80 ± 0.91	90.20 ± 0.24	88.58 ± 0.29	90.10 ± 0.27
Gain	–	$+8.40$	$+6.78$	$+8.30$
CIFAR-100	55.30 ± 0.70	56.66 ± 0.64	58.40 ± 0.68	60.37 ± 0.13
Gain	–	$+1.36$	$+3.11$	$+5.07$

Oracle relation correction produces substantial gains, but the relative importance of FP and FN corruption differs across datasets.

On CIFAR-10, correcting FP relations yields a $+8.40$-pp improvement, larger than the $+6.78$-pp gain from FN recovery. Joint correction reaches 90.10%, close to FP correction alone. This indicates considerable overlap in the damage addressed by the two interventions in this setting.

In contrast, CIFAR-100 exhibits a different pattern. FN recovery provides a larger gain than FP correction ($+3.11$ versus $+1.36$ pp), while jointly correcting both relation types improves accuracy by $+5.07$ pp.

Therefore, we do not claim that one corruption type is universally dominant. Instead, the results show that FP and FN relations represent distinct sources of supervision error whose relative impact depends on the dataset and class structure.

G.2 Relation Identity versus Relation Quantity

The previous experiment shows that oracle relation correction can substantially improve representation quality. However, such gains could in principle arise simply because fewer noisy relations participate in training.

To distinguish which relations are modified from how many relations are modified, we construct a count-matched random intervention.

For each run, we first determine the number of oracle FP relations removed by the targeted intervention. We then remove exactly the same number of observed-positive relations uniformly at random. Thus, the two interventions modify an identical number of relations and differ only in their identity.

Table 19: Oracle FP removal versus count-matched random relation removal.

Dataset	Advantage of targeted FP removal over matched-random removal
CIFAR-10	$+7.61 \pm 0.76$ pp
CIFAR-100	$+4.19 \pm 1.73$ pp

Targeted removal substantially outperforms count-matched random removal on both datasets. Since the intervention size is identical, these differences cannot be explained merely by reducing the amount of pairwise supervision.

Instead, they demonstrate that

> which relations are modified matters, not merely how many relations are removed.

This provides direct evidence for treating the supervision relation, rather than the entire sample, as the relevant unit of intervention.

G.3 Implication for Sample-Level Filtering

Sample deletion modifies all relations incident to a selected example simultaneously. For a sample i, deleting the node removes not only corrupted relations but also potentially useful same-instance information and reliable cross-sample relations.

In contrast, the oracle experiments above intervene only on the relations known to be corrupted while preserving the underlying examples.

The large gap between targeted and count-matched random relation removal therefore supports a finer interpretation of noisy supervision:

$$\text{sample corruption} \not\Rightarrow \text{all relations of that sample are equally harmful.}$$

A mislabeled example can simultaneously participate in useful and corrupted supervision. Consequently, a sample-level keep-or-remove decision is inherently coarser than relation-level modulation.

This observation motivates PRC, which preserves the example and its same-instance positive while continuously reducing the contribution of semantically unsupported label-induced positive relations.

G.4 Summary

The oracle intervention analysis yields two conclusions.

First, both FP and FN relation corruption can substantially degrade representation quality, although their relative importance varies across datasets. Second, controlling for the number of modified relations shows that targeted intervention is substantially more effective than random relation removal.

Together, these findings support the central mechanism claim of the paper:

$$\text{relation identity is a critical determinant of noise harm.}$$

The purpose of PRC is not to reproduce oracle correction, but to approximate this principle using only noisy-data-derived semantic evidence.