Collapse-Aware Regularization for Reliable Reasoning Under Distribution Shift

Quynh T. N. Vo[1,2,3] **Cong-Duy Nguyen**[1]

[1]Center for AI Research, VinUniversity, Vietnam
[2]Ho Chi Minh City University of Technology (HCMUT), Ho Chi Minh City, Vietnam
[3]Vietnam National University Ho Chi Minh City, Ho Chi Minh City, Vietnam

Abstract

Reliable reasoning requires models to generalize under distribution shifts, yet in-distribution validation loss often fails to identify checkpoints that remain reliable out of distribution. We study representation collapse in Transformer hidden states as an internal degeneration that may reduce the effective capacity needed for multi-step inference. We characterize two complementary collapse signals: spectral capacity collapse, measured by the effective rank of layerwise representation covariance, and geometric alignment collapse, measured by average cosine alignment. We then propose the Collapse Risk Criterion (CRC), an ID-computable diagnostic score estimated from in-distribution validation representations. CRC is not a formal OOD guarantee, but a practical surrogate for OOD-free checkpoint selection. Across four reasoning benchmarks and three Transformer backbones, CRC correlates more strongly with OOD reasoning error than ID validation loss and standard confidence-based reliability scores. CRC-aware checkpoint selection improves OOD performance under depth/difficulty and template/rule shifts with minimal ID degradation. As an extension, collapse-aware training with a CRC-based regularizer improves average OOD performance in all evaluated dataset–backbone settings. Our results suggest that monitoring representation collapse is a simple and useful tool for improving reasoning reliability without OOD validation data.

1 INTRODUCTION

Reliable reasoning under distribution shift is a central challenge in modern machine learning. Although Transformer-based reasoners can perform well on in-distribution (ID) validation data, their behavior can degrade substantially under out-of-distribution (OOD) shifts, a setting broadly studied in OOD generalization and domain generalization [Zhou et al., 2023]. This reliability gap is especially important in the common *OOD-free* protocol, where practitioners do not have labeled OOD data for hyperparameter tuning, early stopping, or checkpoint selection. In this setting, checkpoints are typically selected by minimizing ID validation loss, even though this criterion may not reliably identify the checkpoint with the best OOD behavior.

Prior work has approached distribution shift from the perspectives of robust generalization and uncertainty-aware deployment. Surveys on domain generalization and OOD detection emphasize that safe operation requires mechanisms to anticipate or withstand shifts, and that evaluation should go beyond in-distribution metrics [Zhou et al., 2023, Yang et al., 2024]. However, for reasoning models, failures under shift are often subtle: a checkpoint may appear strong on ID validation yet fail on OOD inputs that require longer multi-step inference, use different surface templates, or compose rules in unfamiliar ways. This motivates internal diagnostics that capture failure modes beyond ID loss while remaining computable without OOD labels or OOD examples.

We study *representation collapse* as one such diagnostic signal. Intuitively, if intermediate hidden states become concentrated in a low-dimensional or overly aligned subspace, the model may lose representational diversity needed for robust compositional reasoning. Recent analyses of Transformer representations show that embedding geometry can change substantially across layers and training, including changes in anisotropy and intrinsic dimension [Razzhigaev et al., 2024, Machina and Mercer, 2024]. These observations motivate using spectral and geometric properties of hidden states as ID-computable indicators of collapse risk.

Building on this perspective, we characterize two complementary collapse signals. First, *spectral capacity collapse* is measured by the effective rank of the layerwise representation covariance, capturing whether variation concen-

Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence (UAI 2026), PMLR 337:6999–7022.

trates into a small number of active directions. Second, *geometric alignment collapse* is measured by average cosine alignment, capturing whether representations become overly collinear. We also track representation variance as a diagnostic quantity. From these signals, we define the *Collapse Risk Criterion* (CRC), an ID-computable diagnostic score that summarizes layerwise collapse risk. CRC is not intended as a formal OOD guarantee; rather, it is a practical surrogate for ranking checkpoints when OOD validation data are unavailable.

Our approach targets the *OOD-free reliability* protocol: all training, threshold calibration, hyperparameter selection, and checkpoint selection use only ID data, while OOD sets are used exclusively for final evaluation. We show that CRC provides a stronger checkpoint-level signal of OOD reasoning error than ID validation loss and standard reliability baselines. More importantly, CRC enables OOD-free checkpoint selection: selecting checkpoints using ID validation loss augmented with CRC improves OOD performance under depth/difficulty and template/rule shifts, with only small changes in ID validation accuracy. We further study collapse-aware training as an extension, showing that a CRC-based regularizer can reduce representation degeneration and improve average OOD performance across the evaluated dataset–backbone grid.

In summary, we make the following contributions:

- We formalize two representation-level collapse signals for Transformer reasoning: *spectral capacity collapse*, measured by effective rank, and *geometric alignment collapse*, measured by cosine alignment.
- We introduce the *Collapse Risk Criterion* (CRC), an ID-computable diagnostic score that aggregates layerwise collapse statistics without using OOD labels or OOD examples.
- We show that CRC enables OOD-free checkpoint selection. Across reasoning datasets and Transformer backbones, CRC is more predictive of OOD error than ID validation loss and standard reliability baselines, and CRC-aware selection improves OOD performance under depth/difficulty and template/rule shifts.
- We study collapse-aware training as an extension. A CRC-based regularizer improves average OOD performance in all 12 evaluated dataset–backbone settings while preserving the OOD-free protocol.

2 PROBLEM SETTING

2.1 OOD-FREE RELIABILITY FOR REASONING

We consider a supervised reasoning task with inputs $x \in \mathcal{X}$ and outputs $y \in \mathcal{Y}$. Let P denote the in-distribution (ID) joint distribution used for training and validation, and let Q denote an out-of-distribution (OOD) test distribution induced by reasoning shifts. In the *OOD-free* protocol studied in this paper, samples from Q are not available during training, hyperparameter tuning, early stopping, threshold selection, or checkpoint selection.

We observe ID training and validation sets

$$\begin{aligned} D_{\mathrm{tr}} &= \{(x_i, y_i)\}_{i=1}^{N_{\mathrm{tr}}} \sim P, \\ D_{\mathrm{val}} &= \{(x_i, y_i)\}_{i=1}^{N_{\mathrm{val}}} \sim P, \end{aligned} \tag{1}$$

and one or more OOD test sets

$$D_{\mathrm{ood}} = \{(x_i, y_i)\}_{i=1}^{N_{\mathrm{ood}}} \sim Q, \tag{2}$$

which are used exclusively for final evaluation. Let $f_\theta : \mathcal{X} \to \mathcal{Y}$ be a Transformer-based reasoner and let ℓ denote the task loss. For any distribution $\mathcal{D} \in \{P, Q\}$, the population risk is

$$R_{\mathcal{D}}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\ell(f_\theta(x), y)\right]. \tag{3}$$

2.2 OOD-FREE CHECKPOINT SELECTION

Training produces a sequence of checkpoints $\{\theta_t\}_{t=1}^{T}$. The standard ID-only selection rule chooses

$$t^\star_{\mathrm{ID}} = \arg\min_{t\in\{1,\dots,T\}} R_{D_{\mathrm{val}}}(\theta_t), \tag{4}$$

where $R_{D_{\mathrm{val}}}$ is the empirical validation risk. This rule can select checkpoints that perform well on ID validation but generalize poorly under reasoning shifts.

We study OOD-free checkpoint selection: choosing a checkpoint $t^\star$ using only ID information, with the goal of achieving lower OOD risk than ID-loss selection:

$$R_Q(\theta_{t^\star}) < R_Q(\theta_{t^\star_{\mathrm{ID}}}) \quad \text{in final OOD evaluation.} \tag{5}$$

This inequality is a target evaluation criterion rather than a formal guarantee. Our method, CRC, provides an ID-computable collapse-risk criterion for ranking checkpoints without accessing OOD data.

2.3 REASONING SHIFTS

We focus on OOD distributions Q that preserve the task definition but alter reasoning demands or input structure. We consider three shift families. First, *difficulty/depth shifts* increase the number of inference steps or compositional complexity, e.g., moving from short proofs to longer multi-hop chains. Second, *template or surface-form shifts* change phrasing, formatting, symbol order, or input templates while preserving semantics. Third, *rule or composition shifts* alter how relations or inference rules are combined, testing whether the model has learned compositional reasoning rather than frequent ID patterns.

2.4 EVALUATION PROTOCOL

We report ID and OOD task performance:

$$\begin{aligned} \mathrm{Acc}_{\mathrm{ID}}(\theta) &= \mathrm{Acc}(f_\theta; D_{\mathrm{val}}), \\ \mathrm{Acc}_{\mathrm{OOD}}(\theta) &= \mathrm{Acc}(f_\theta; D_{\mathrm{ood}}). \end{aligned} \tag{6}$$

or analogous generation metrics such as exact match. The goal is to improve OOD performance without substantially degrading ID performance. All model-selection and hyperparameter decisions are made using ID data only; OOD sets are reserved strictly for final reporting. When applicable, we average results over multiple random seeds.

We also report checkpoint-level predictiveness using Spearman rank correlation between ID-computable risk signals and final OOD error. This evaluates whether a score can rank checkpoints by their failure risk beyond average ID validation accuracy.

3 METHOD

We propose the *Collapse Risk Criterion* (CRC), an ID-computable diagnostic for ranking checkpoints by representation-collapse risk. CRC is designed for the OOD-free reliability setting in §2: checkpoint statistics, thresholds, and selection decisions are computed using ID validation data only, while the training-time regularizer uses ID training minibatches. No OOD examples or OOD labels are used in either case. CRC should be interpreted as an empirical collapse-risk criterion for checkpoint selection, not as a formal OOD-risk guarantee.

3.1 LAYERWISE REPRESENTATIONS

Let f_θ be a Transformer-based reasoner. Given an input x, we denote the hidden states at layer ℓ by

$$H^\ell(x;\theta) \in \mathbb{R}^{n\times d}, \tag{7}$$

where n is the number of token or pooled positions and d is the hidden dimension. To compute layerwise collapse statistics, we collect a set of m representation vectors

$$\mathcal{Z}_\theta^\ell = \{z_j^\ell\}_{j=1}^m, \qquad z_j^\ell \in \mathbb{R}^d, \tag{8}$$

from the ID validation set. Depending on the backbone, z_j^ℓ is either a pooled sequence representation, such as a `[CLS]` state or mean-pooled encoder state, or a token-level representation. We keep this choice fixed for each backbone and report the exact extraction rule in Appendix C.1.

For each layer ℓ, we compute the empirical covariance

$$\begin{aligned} \bar{z}^\ell &= \frac{1}{m}\sum_{j=1}^m z_j^\ell, \\ C^\ell(\theta) &= \frac{1}{m-1}\sum_{j=1}^m \left(z_j^\ell - \bar{z}^\ell\right)\left(z_j^\ell - \bar{z}^\ell\right)^\top. \end{aligned} \tag{9}$$

All collapse statistics below are computed per layer and then aggregated over a monitored layer set $\mathcal{S}$.

3.2 TWO MODES OF REPRESENTATION COLLAPSE

We view reasoning collapse as a representation-level degeneration that reduces the diversity or separability of hidden states needed for multi-step inference. We operationalize collapse through two complementary signals: spectral collapse and geometric alignment collapse.

Spectral collapse. Let $\{\lambda_i^\ell\}_{i=1}^d$ be the eigenvalues of $C^\ell(\theta)$. We normalize the spectrum as

$$p_i^\ell = \frac{\lambda_i^\ell}{\sum_{j=1}^d \lambda_j^\ell + \varepsilon}, \tag{10}$$

where $\varepsilon > 0$ is a small numerical constant. We then compute the effective rank:

$$r_{\mathrm{eff}}^\ell(\theta) = \exp\left(-\sum_{i=1}^d p_i^\ell \log p_i^\ell\right). \tag{11}$$

The effective rank estimates how many directions are actively used in the representation space. A lower r_{eff}^ℓ indicates that variation concentrates into a smaller subspace, suggesting spectral collapse.

Geometric alignment collapse. To measure whether representations become overly collinear, we ℓ_2-normalize each vector,

$$\tilde{z}_j^\ell = \frac{z_j^\ell}{\|z_j^\ell\|_2 + \varepsilon}, \tag{12}$$

and compute the average pairwise cosine similarity:

$$c^\ell(\theta) = \frac{1}{m(m-1)}\sum_{i\neq j}\left\langle \tilde{z}_i^\ell, \tilde{z}_j^\ell \right\rangle. \tag{13}$$

Higher c^ℓ means that representations are more aligned. We also measure the total spread of the representation cloud using the covariance trace:

$$v^\ell(\theta) = \mathrm{tr}\big(C^\ell(\theta)\big) = \sum_{i=1}^d \lambda_i^\ell. \tag{14}$$

Geometric collapse corresponds to increased cosine alignment and reduced spread, indicating that hidden states become less diverse.

Complementarity of the signals. Although effective rank and cosine alignment are related, they capture different aspects of collapse. Effective rank measures how covariance energy is distributed across centered spectral directions, whereas average cosine similarity measures global directional alignment after normalization. A representation cloud

can have a dominant common direction while retaining nontrivial centered variance, or can have reduced spectral diversity without uniformly high pairwise cosine similarity. We therefore combine both signals to capture complementary collapse modes.

3.3 COLLAPSE RISK CRITERION

CRC aggregates layerwise collapse statistics into a scalar diagnostic score for checkpoint ranking. Let $\mathcal{S}$ be the monitored layer set. We define

$$\mathrm{CRC}(\theta) = \frac{1}{|\mathcal{S}|} \sum_{\ell \in \mathcal{S}} \Big[\lambda_c \, \phi_c(c^\ell(\theta)) + \lambda_r \, \phi_r(r^\ell_{\mathrm{eff}}(\theta)) + \lambda_v \, \phi_v(v^\ell(\theta)) \Big]. \tag{15}$$

where $\lambda_c, \lambda_r, \lambda_v \geq 0$ are component weights. The hinge penalties are

$$\begin{aligned} \phi_c(c) &= \max(0, c - \tau_c), \\ \phi_r(r) &= \max(0, \tau_r - r), \\ \phi_v(v) &= \max(0, \tau_v - v). \end{aligned} \tag{16}$$

The thresholds τ_c, τ_r, τ_v are calibrated using ID validation representations only, for example through layerwise percentile rules. No OOD data is used to set these thresholds. In our default CRC, we use $\lambda_c = 1$, $\lambda_r = 1$, and $\lambda_v = 0$; the variance component is reported for diagnostic and ablation variants. This choice reflects our empirical finding that cosine alignment and effective rank provide the most stable checkpoint-selection signal, while variance is useful for analysis.

CRC penalizes excessive alignment and insufficient effective capacity, thereby assigning higher scores to checkpoints whose ID representations show stronger collapse. Unlike ID validation loss, CRC is sensitive to internal degeneration that may not immediately reduce ID accuracy but can make the model brittle under deeper or shifted reasoning inputs.

3.4 OOD-FREE CHECKPOINT SELECTION

Training produces a sequence of checkpoints $\{\theta_t\}_{t=1}^T$. The standard ID-only rule selects

$$t^\star_{\mathrm{ID}} = \arg \min_{t \in \{1,\ldots,T\}} R_{D_{\mathrm{val}}}(\theta_t), \tag{17}$$

which may not select the most robust checkpoint under OOD reasoning shifts.

CRC provides an alternative ID-only checkpoint-selection criterion. We use a collapse-aware score

$$S_{\mathrm{CRC}}(\theta_t) = R_{D_{\mathrm{val}}}(\theta_t) + \eta \, \mathrm{CRC}(\theta_t), \tag{18}$$

Algorithm 1: OOD-Free Checkpoint Selection with CRC

Input: ID validation set D_{val}; checkpoints $\{\theta_t\}_{t=1}^T$; monitored layers $\mathcal{S}$; ID-calibrated thresholds τ_c, τ_r, τ_v; weights $\lambda_c, \lambda_r, \lambda_v$; tradeoff η
Output: Selected checkpoint $\theta_{t^\star_{\mathrm{CRC}}}$
for $t = 1$ **to** T **do**
  Compute ID validation risk $R_{D_{\mathrm{val}}}(\theta_t)$;
  Initialize $\mathrm{CRC}(\theta_t) \leftarrow 0$;
  foreach $\ell \in \mathcal{S}$ **do**
    Collect ID representations $\mathcal{Z}^\ell_{\theta_t}$ from D_{val};
    Compute layerwise collapse statistics $c^\ell(\theta_t)$, $r^\ell_{\mathrm{eff}}(\theta_t)$, and $v^\ell(\theta_t)$ using Eqs. (11)–(14);
    Add the weighted hinge penalties in Eq. (15) to $\mathrm{CRC}(\theta_t)$;
  Normalize $\mathrm{CRC}(\theta_t)$ by $|\mathcal{S}|$;
  Compute $S_{\mathrm{CRC}}(\theta_t) = R_{D_{\mathrm{val}}}(\theta_t) + \eta \, \mathrm{CRC}(\theta_t)$;
$t^\star_{\mathrm{CRC}} \leftarrow \arg\min_t S_{\mathrm{CRC}}(\theta_t)$;
return $\theta_{t^\star_{\mathrm{CRC}}}$;

where $\eta \geq 0$ controls the strength of collapse-aware selection. The tradeoff η is fixed using ID validation data only and is not tuned on OOD results. The selected checkpoint is

$$t^\star_{\mathrm{CRC}} = \arg \min_{t \in \{1,\ldots,T\}} S_{\mathrm{CRC}}(\theta_t). \tag{19}$$

We also report a Pareto-style variant that avoids a continuous tradeoff: among the top-k checkpoints by ID validation loss, we select the checkpoint with the lowest CRC. Both variants are OOD-free: OOD test sets are used only for final evaluation and never for tuning, thresholding, or checkpoint selection.

3.5 COLLAPSE-AWARE TRAINING OBJECTIVE

As an extension, we also study whether CRC can be used as a training-time regularizer. Instead of only ranking checkpoints after training, we add a collapse penalty to the task objective:

$$\min_\theta \; \mathbb{E}_{(x,y)\sim D_{\mathrm{tr}}} \left[\ell(f_\theta(x), y) \right] + \alpha \, \mathrm{CRC}(\theta), \tag{20}$$

where $\alpha \geq 0$ is selected using ID validation data only. In practice, we compute CRC on minibatch representations and restrict the monitored layer set $\mathcal{S}$ to a small subset of layers to control overhead.

This objective discourages excessive representation collapse while preserving the ID task objective. We treat collapse-aware training as a secondary extension: the central contribution of this paper is OOD-free checkpoint selection with CRC, while the training objective tests whether the same collapse signal can also improve robustness during optimization.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

We evaluate CRC in the *OOD-free reliability* setting: all training, hyperparameter tuning, threshold calibration, and checkpoint selection use only in-distribution (ID) data. Out-of-distribution (OOD) sets are used exclusively for final evaluation. For each dataset, we construct an ID split $(D_{\mathrm{tr}}, D_{\mathrm{val}}) \sim P$ and two OOD test sets, $D_{\mathrm{test}}^{(A)}$ and $D_{\mathrm{test}}^{(B)}$, corresponding to two reasoning-shift families. We report mean and standard deviation over multiple random seeds.

Datasets. We use four logical reasoning benchmarks spanning entailment-style and proof-style supervision: LogicNLI [Tian et al., 2021], ProntoQA [Saparov and He, 2023], ProofWriter [Tafjord et al., 2021], and ProverQA [Qi et al., 2025]. All tasks are cast into a unified text-to-label format. Multi-class tasks use a classification head with cross-entropy loss, while binary provability tasks use binary classification. Full preprocessing details and label mappings are provided in Appendix A.

OOD shifts. For each dataset we construct two OOD test sets that preserve the task semantics: **OOD-A** increases reasoning depth or difficulty (hop count, proof length, or logical complexity), and **OOD-B** perturbs surface templates or rule-composition patterns. Table 1 summarizes the construction; split sizes and preprocessing rules are in Appendix A.2.

Backbone models. We evaluate three Transformer backbones: DeBERTa-v3 [He et al., 2021], Longformer [Beltagy et al., 2020], and T5 [Raffel et al., 2020]. DeBERTa-v3 and Longformer are used as encoder-only classifiers. T5 is used in text-to-text format, and CRC is computed on encoder representations to avoid decoder-side confounds and to keep the diagnostic comparable across backbones.

Selection rules. Training produces checkpoints $\{\theta_t\}_{t=1}^{T}$. We compare: (i) **Select-ID**, which selects the checkpoint with the lowest ID validation loss; (ii) **Select-CRC**, which selects the checkpoint minimizing $R_{D_{\mathrm{val}}}(\theta_t) + \eta\,\mathrm{CRC}(\theta_t)$; and (iii) **Pareto+CRC**, which first keeps the top-k checkpoints by ID validation loss and then selects the one with the lowest CRC. All selection hyperparameters are chosen using ID data only.

CRC computation. Unless otherwise stated, CRC is computed from pooled ID validation representations. For DeBERTa-v3 and Longformer, we use `[CLS]` representations; for T5, we use mean-pooled encoder representations. The default monitored layers are mid-to-late layers. The default CRC uses $\lambda_c = 1$, $\lambda_r = 1$, and $\lambda_v = 0$; the variance component is reported for diagnostic and ablation variants. Thresholds are calibrated from ID validation statistics using the 90/10 percentile rule. Full implementation details are in Appendix C.

Metrics. We report ID validation accuracy and OOD accuracy on both shifts. We also evaluate checkpoint-level predictiveness using Spearman correlation between risk signals and OOD error. For additional reliability analysis, Appendix F reports full checkpoint-level predictiveness results.

4.2 MAIN RESULT: OOD-FREE CHECKPOINT SELECTION

Table 2 summarizes OOD-free checkpoint selection over the full 4 datasets $\times$ 3 backbones grid. Compared with Select-ID, Select-CRC improves OOD performance across both shifts while incurring only a small ID-validation trade-off. This supports CRC as an ID-computable checkpoint-ranking criterion for reasoning robustness. The full dataset-by-backbone matrix is reported in Appendix E.

4.3 DOES CRC PREDICT OOD FAILURE?

For each training run, we compute CRC on ID validation representations at every checkpoint and measure how well each signal ranks checkpoints by their final OOD error using Spearman rank correlation. We compare CRC against five ID-computable baselines that capture distinct sources of checkpoint-quality information: **ID validation loss**, the standard model-selection signal; **Best confidence baseline**, the strongest of maximum softmax probability, predictive entropy, and energy per setting; **Linear probe (ID val)**, a representation-level baseline that trains a logistic regression on pooled hidden states at the CRC-monitored layer and scores by 1 minus its 5-fold cross-validated accuracy; **Ensemble disagreement**, the fraction of ID validation examples on which the multi-seed predictions disagree; and **ATC** [Garg et al., 2022], an adaptation of average-threshold confidence as a checkpoint-level signal. Together, these baselines span output-level, representation-level, and ensemble-level reliability signals. Full implementation details are in Appendices D and F.1.

Table 3 reports the range of checkpoint-level Spearman correlations across datasets and backbones, and Appendix F reports the full per-setting matrix. CRC achieves the strongest correlation with OOD error in this comparison. The linear probe outperforms ID validation loss and confidence-style baselines but remains below CRC, suggesting that representation-collapse geometry captures checkpoint-ranking information beyond linear separability of ID labels. Ensemble disagreement is a competitive baseline when multiple seeds are available, but does not consistently match CRC. The ATC adaptation (Appendix F.1) also improves over confidence-style baselines but remains below CRC. Together, these comparisons indicate that representation-collapse geometry contains checkpoint-ranking information not fully captured by output-level or representation-level baselines.

Table 1: OOD shift construction. OOD sets are used only for final evaluation, never for training, tuning, or model selection.

Dataset	OOD-A: depth/difficulty	OOD-B: template/rule	Label preservation
LogicNLI	Logical complexity (facts/rules/ops)	Template/rule reordering	Same label space
ProntoQA	Native hop count	Template/order perturbation	True/false unchanged
ProofWriter	Proof complexity (facts/rules/chains)	Rule/fact reordering	Same target and label
ProverQA	Native easy/medium/hard difficulty	Rule/template variation	Same answer field

Table 2: OOD-free checkpoint selection summary over 4 datasets $\times$ 3 backbones. Δ denotes Select-CRC minus Select-ID.

Dataset	# settings	Δ ID	Δ OOD-A	Δ OOD-B
LogicNLI	3	-0.01	+0.04	+0.04
ProntoQA	3	-0.01	+0.05	+0.04
ProofWriter	3	-0.01	+0.06	+0.06
ProverQA	3	-0.01	+0.06	+0.06
Overall	**12**	**-0.01**	**+0.05**	**+0.05**

Table 3: Checkpoint-level correlation with OOD error across datasets and backbones. Higher Spearman ρ indicates stronger predictiveness.

Signal	Spearman ρ range
ID validation loss	0.24–0.36
Best confidence baseline	0.33–0.46
Linear probe (ID val)	0.44–0.58
ATC [Garg et al., 2022]	0.52–0.58
Ensemble disagreement	0.52–0.68
CRC (ours)	**0.60–0.80**

4.4 DOES CRC TRACK REASONING DIFFICULTY?

To distinguish reasoning robustness from generic shift robustness, we analyze OOD-A performance by reasoning complexity. For ProntoQA and ProverQA, we use native metadata such as hop count and difficulty labels. For ProofWriter and LogicNLI, where uniform depth labels are unavailable in the processed splits, we use deterministic structural proxies based on the number of facts, rules, logical operators, and rule/fact composition patterns. Table 4 shows that CRC gains increase with reasoning complexity across datasets.

Because reasoning depth can correlate with input length, we further conduct a length-matched control. For each dataset, we stratify OOD-A examples into tokenized input-length bins and construct matched subsets with comparable length distributions across depth groups. As shown in Table 11 in Appendix B, the increasing-gain trend remains after matching, reducing the likelihood that CRC is merely selecting checkpoints that perform better on longer inputs.

Table 4: CRC gains increase with reasoning complexity on OOD-A. Gains are Select-CRC minus Select-ID.

Dataset	Shallow/easy	Medium	Deep/hard
ProntoQA	+0.02 (1-hop)	+0.04 (2-hop)	+0.07 (3–4-hop)
ProverQA	+0.03 (Easy)	+0.05 (Medium)	+0.08 (Hard)
ProofWriter	+0.02 (depth 1)	+0.06 (depth 3)	+0.09 (depth 5)
LogicNLI	+0.02 (depth 1)	+0.04 (depth 3)	+0.07 (depth 5)

Table 5: Threshold sensitivity on ProverQA–DeBERTa.

Rule	OOD-A	OOD-B	Spearman ρ
Select-ID	0.54	0.50	0.31
CRC 85/15	0.59	0.55	0.71
CRC 90/10	0.60	0.56	0.73
CRC 95/5	0.59	0.55	0.72

Sensitivity to selection and training coefficients. We further examine whether CRC depends on finely tuned coefficient choices. Figure 1 shows that Select-CRC is stable across a moderate range of η, while CRC-training exhibits a clearer ID–OOD trade-off as α varies.

4.5 ROBUSTNESS TO CRC DESIGN CHOICES

We next test whether CRC depends on brittle design choices. Table 5 reports an ID-only threshold sweep on ProverQA–DeBERTa. CRC remains stable around the default 90/10 rule, especially from 85/15 to 95/5, indicating that performance is not driven by a single hand-picked percentile. Appendix G further reports monitored-layer sensitivity and component ablations, showing that CRC is strongest on mid-to-late layers and mainly driven by effective-rank and cosine-alignment signals.

For monitored layers, mid-to-late representations perform best, but the trend is gradual rather than brittle. This suggests that CRC is not dependent on one fragile layer choice, while also supporting the intuition that collapse becomes most predictive after intermediate reasoning composition.

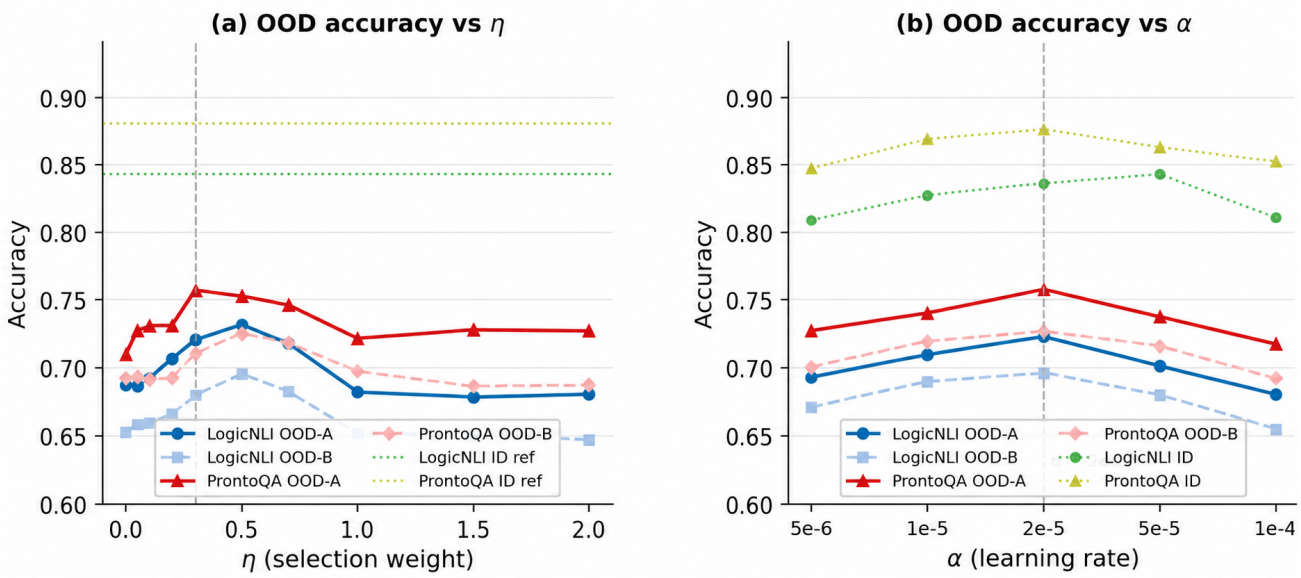

Figure 1: Sensitivity to CRC coefficients. Panel (a) varies the selection weight η, and panel (b) varies the training coefficient α. Dashed lines mark default values; dotted lines show ID reference performance.

Table 6: CRC-training extension averaged over three backbones for each dataset. Δ denotes CRC-train minus standard fine-tuning under the same ID-only checkpoint-selection protocol.

Dataset	Δ ID	Δ Avg. OOD	**Wins**	**# settings**
LogicNLI	-0.01	+0.04	3/3	3
ProntoQA	-0.01	+0.05	3/3	3
ProofWriter	-0.01	+0.06	3/3	3
ProverQA	-0.01	+0.06	3/3	3
Overall	**-0.01**	**+0.05**	**12/12**	**12**

4.6 CRC-TRAINING AS AN EXTENSION

Although our central contribution is OOD-free checkpoint selection, we also study whether the same collapse signal can regularize training. We compare standard fine-tuning against collapse-aware fine-tuning with $\mathcal{L}_{\text{task}} + \alpha\,\text{CRC}$, where α is selected using ID validation data only. To isolate the effect of training regularization from checkpoint-selection bias, both the standard and CRC-regularized trajectories use the same Select-ID rule within their respective training runs. Thus, differences in Table 6 reflect the training objective rather than a different selection rule.

Table 6 summarizes the results averaged over backbones for each dataset, with the full matrix reported in Appendix H. CRC-training improves Avg. OOD performance in all 12 dataset–backbone settings, with only a small average ID-validation change. This supports collapse-aware training as a useful extension of CRC, while OOD-free checkpoint selection remains the central and most direct contribution.

Layerwise collapse dynamics. To examine whether the robustness gains are associated with reduced representation collapse, Figure 2 plots the layerwise effective rank, average cosine similarity, and trace variance for ID, OOD-baseline, and OOD CRC-trained representations. OOD examples exhibit a clear collapse pattern in mid-to-late layers: effective rank decreases, cosine alignment increases, and trace variance decreases. CRC-training partially reverses all three trends, supporting the interpretation that, in settings where CRC-training helps, the gains are associated with reduced representation degeneration.

5 RELATED WORKS

5.1 OOD GENERALIZATION AND RELIABILITY UNDER SHIFT

A central theme in modern machine learning is how to maintain performance and make reliable decisions when test data differ from the training distribution. Early work formalized this challenge through *distributionally robust optimization* (DRO), which seeks models that perform well under adversarially chosen shifts within a specified uncertainty set [Ben-Tal et al., 2009]. More recently, *domain generalization* methods aim to learn invariances across multiple source environments so that the learned predictor transfers to unseen domains [Lai and Wang, 2024, Zhou et al., 2023]. In parallel, the OOD literature has developed OOD detection and uncertainty-estimation methods, including confidence-based baselines [Hendrycks and Gimpel, 2017], calibrated scoring rules such as ODIN [Liang et al., 2018], energy-based scores [Liu et al., 2020], and Bayesian or ensemble-based

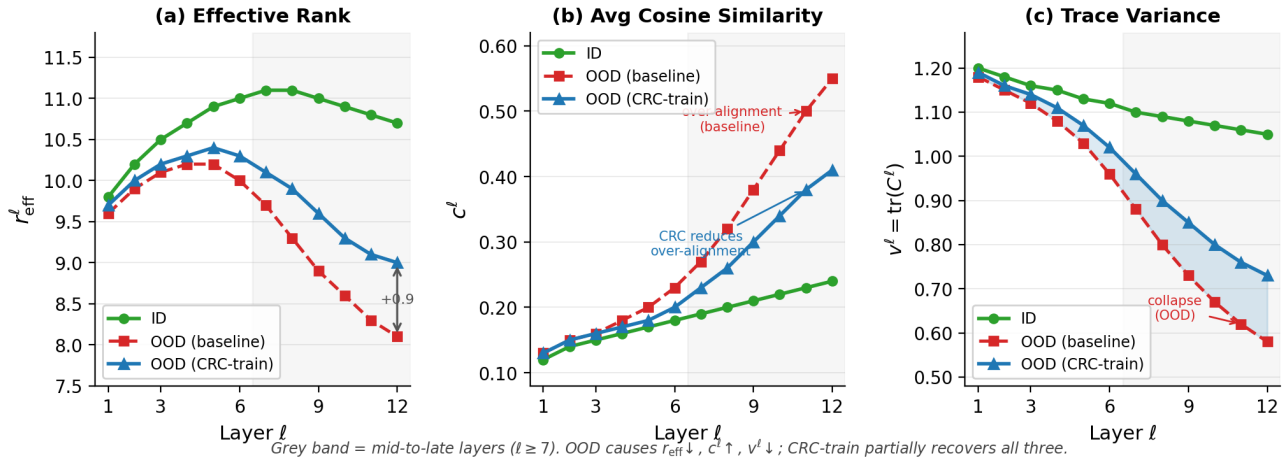

Figure 2: Layerwise representation-collapse diagnostics. The grey band marks mid-to-late layers where OOD examples show lower effective rank, higher cosine alignment, and lower trace variance. CRC-training partially reverses these trends, suggesting reduced representation degeneration.

uncertainty estimates [Gal and Ghahramani, 2016, Lakshminarayanan et al., 2017]. A complementary reliability perspective is *selective prediction*, where a model abstains on high-risk inputs to control error under shift [Geifman and El-Yaniv, 2017].

Our work targets a different but practical setting: *OOD-free* reliability for reasoning. During training, threshold calibration, hyperparameter selection, and checkpoint selection, we only observe ID data; OOD sets are reserved strictly for final evaluation. Unlike DRO or IRM-style objectives that require explicit shift modeling or multiple environments, and unlike standard OOD detection methods that typically operate on logits or input-level confidence scores, we use an internal diagnostic based on layerwise representation geometry. Specifically, the Collapse Risk Criterion (CRC) aggregates spectral and geometric collapse signals from hidden states to support OOD-free checkpoint selection, and as an extension, collapse-aware training. This positions CRC as a practical ID-computable reliability tool that complements existing OOD generalization, uncertainty estimation, and confidence-based selection methods.

5.2 REPRESENTATION COLLAPSE, ANISOTROPY, AND GEOMETRY OF TRANSFORMER FEATURES

A growing body of work suggests that the geometry of learned representations, including anisotropy and dimensionality, is closely tied to downstream behavior and robustness. Foundational analyses of sentence embeddings highlighted anisotropy and proposed post-processing methods to improve embedding geometry [Mu and Viswanath, 2018], while subsequent studies quantified anisotropy across Transformer layers and tasks [Ethayarajh, 2019]. More recent investigations examine how representation geometry evolves during training, including changes in intrinsic dimension and layerwise spectral structure [Razzhigaev et al., 2024], and argue that anisotropy depends on objectives and training dynamics rather than being an unavoidable property of Transformer representations [Machina and Mercer, 2024]. Separately, contrastive learning and embedding-learning work has connected collapse, uniformity, and alignment trade-offs to representation quality and generalization [Wang and Isola, 2020].

While prior work often studies anisotropy or collapse as a representational phenomenon, our goal is to use collapse as an ID-computable reliability signal for reasoning under distribution shift. We characterize two complementary collapse signals: spectral capacity collapse, measured by effective rank, and geometric alignment collapse, measured by average cosine alignment. We also track trace variance as a diagnostic quantity and ablation component. These signals are combined into CRC, which we use as a checkpoint-ranking criterion in the OOD-free protocol and as a training-time regularizer in an extension. Thus, rather than treating representation collapse only as an embedding-quality issue to be corrected post hoc, we use it as an internal diagnostic for selecting and training more reliable reasoning models under structured reasoning shifts.

6 DISCUSSION AND LIMITATIONS

Our results suggest that representation collapse is a useful internal signal for improving the reliability of Transformer-based reasoning under distribution shift. In the OOD-free setting, CRC is computed only from ID validation representations, yet it provides stronger checkpoint-level predictiveness of OOD error than ID validation loss, output-

confidence baselines, linear-probe diagnostics, ATC, and ensemble disagreement (Table 3). This predictiveness translates into practical gains: CRC-aware checkpoint selection improves OOD performance across the full 4 datasets $\times$ 3 backbones grid under both depth/difficulty and template/rule shifts, while incurring only a small ID-validation trade-off (Table 2 and Appendix E). The depth-binned analysis further shows that CRC gains increase with reasoning complexity, suggesting that the diagnostic is particularly useful when examples require deeper or more compositional inference (Table 4). Finally, collapse-aware training provides a useful extension: adding a CRC-based regularizer improves average OOD performance in all 12 evaluated dataset–backbone settings, although checkpoint selection remains the most direct use of CRC because it does not require modifying training.

Why does CRC help? A consistent empirical pattern is that weaker OOD checkpoints exhibit lower effective rank and stronger cosine alignment in mid-to-late layers. This suggests that their hidden representations concentrate into fewer active directions and become more collinear, reducing the representational diversity needed for multi-step reasoning. Such degeneration may not be fully reflected in ID validation loss, because a checkpoint can remain locally strong on familiar ID patterns while becoming brittle under deeper or compositionally shifted inputs. CRC helps by ranking checkpoints not only by ID task fit, but also by the health of their internal representation geometry. This explains why CRC-aware selection can prefer checkpoints with slightly higher ID loss but better OOD robustness.

Limitations. CRC should be interpreted as an empirical, ID-computable collapse-risk diagnostic, not as a formal OOD guarantee. Although CRC correlates with OOD error and improves checkpoint selection in our evaluated settings, it does not provide a distribution-free bound on $R_Q(\theta)$ without additional assumptions. Our evaluation also focuses on reasoning-relevant shifts, especially depth/difficulty and template/rule variation. Other shift families, such as domain shifts, noisy inputs, adversarial perturbations, or long-context distribution shifts, may expose failure modes not captured by the current diagnostic. CRC also introduces design choices, including the monitored layer set, percentile thresholds, active collapse components, and the selection or training coefficients. Our sensitivity analyses show that CRC is stable around the default choices (Table 5 and Appendix G), but more automated or theoretically grounded calibration remains an open direction. Finally, CRC requires access to hidden states and covariance statistics. We reduce the overhead by using pooled representations and mid-to-late layers, but scaling the diagnostic to very large models or extremely long contexts may require more efficient estimators.

Future Work. Future work can extend CRC in three directions. First, CRC could be evaluated under broader shift families and on larger generative reasoners. Second, causal interventions on layer subsets could test whether reducing collapse directly improves reasoning robustness, rather than merely correlating with it. Third, more efficient online estimators could make collapse monitoring practical during large-scale training or deployment. Overall, our findings support a practical reliability direction: when OOD validation data are unavailable, model selection and training can benefit from monitoring internal representation health in addition to ID validation loss.

7 CONCLUSION

We studied the reliability of Transformer reasoning under distribution shift through the lens of representation collapse. We formalized two complementary collapse signals: spectral capacity collapse, measured by effective rank, and geometric alignment collapse, measured by cosine alignment. Building on these signals, we proposed the Collapse Risk Criterion (CRC), an ID-computable diagnostic score for estimating collapse risk from validation representations without using OOD data. Across four reasoning benchmarks and three Transformer backbones, CRC provides a stronger checkpoint-level signal of OOD error than ID validation loss and standard reliability baselines. CRC-aware checkpoint selection improves OOD performance under depth/difficulty and template/rule shifts with only small changes in ID validation accuracy. We further studied collapse-aware training as an extension and found that a CRC-based regularizer can provide additional OOD gains. Overall, our results suggest that monitoring representation geometry offers a simple and practical tool for improving reasoning reliability when OOD validation data are unavailable.

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer, 2020. URL https://arxiv.org/abs/2004.05150.

Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. 2009.

Kawin Ethayarajh. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1006. URL https://aclanthology.org/D19-1006/.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, page 1050–1059. JMLR.org, 2016.

Saurabh Garg, Sivaraman Balakrishnan, Zachary Chase Lipton, Behnam Neyshabur, and Hanie Sedghi. Leveraging unlabeled data to predict out-of-distribution performance. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=o_HsiMPYh_x.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/4a8423d5e91fda00bb7e46540e2b0cf1-Paper.pdf.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. {DEBERTA}: {DECODING}-{enhanced} {bert} {with} {disentangled} {attention}. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=XPZIaotutsD.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=Hkg4TI9xl.

Zhao-Rong Lai and Weiwen Wang. Invariant risk minimization is a total variation model. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6405–6416, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=H1VGkIxRZ.

Weitang Liu, Xiaoyun Wang, John D. Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Anemily Machina and Robert Mercer. Anisotropy is not inherent to transformers. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4892–4907, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.274. URL https://aclanthology.org/2024.naacl-long.274/.

Jiaqi Mu and Pramod Viswanath. All-but-the-top: Simple and effective postprocessing for word representations. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=HkuGJ3kCb.

Chengwen Qi, Ren Ma, Bowen Li, He Du, Binyuan Hui, Jinwang Wu, Yuanjun Laili, and Conghui He. Large language models meet symbolic provers for logical reasoning evaluation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=C25SgeXWjE.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Anton Razzhigaev, Matvey Mikhalchuk, Elizaveta Goncharova, Ivan Oseledets, Denis Dimitrov, and Andrey Kuznetsov. The shape of learning: Anisotropy and intrinsic dimensions in transformer-based models. In Yvette Graham and Matthew Purver, editors, *Findings of the Association for Computational Linguistics: EACL 2024*, pages 868–874, St. Julian's, Malta, March 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-eacl.58. URL https://aclanthology.org/2024.findings-eacl.58/.

Abulhair Saparov and He He. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=qFVVBzXxR2V.

Oyvind Tafjord, Bhavana Dalvi, and Peter Clark. ProofWriter: Generating implications, proofs, and abductive statements over natural language. In Chengqing

Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3621–3634, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.317. URL https://aclanthology.org/2021.findings-acl.317/.

Jidong Tian, Yitian Li, Wenqing Chen, Liqiang Xiao, Hao He, and Yaohui Jin. Diagnosing the first-order logical reasoning ability through LogicNLI. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3738–3747, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.303. URL https://aclanthology.org/2021.emnlp-main.303/.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9929–9939. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/wang20k.html.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *Int. J. Comput. Vision*, 132(12):5635–5662, June 2024. ISSN 0920-5691. doi: 10.1007/s11263-024-02117-4. URL https://doi.org/10.1007/s11263-024-02117-4.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4396–4415, 2023. doi: 10.1109/TPAMI.2022.3195549.

Collapse-Aware Regularization for Reliable Reasoning Under Distribution Shift (Supplementary Material)

Quynh T. N. Vo[1,2,3] **Cong-Duy Nguyen**[1]

[1]Center for AI Research, VinUniversity, Vietnam
[2]Ho Chi Minh City University of Technology (HCMUT), Ho Chi Minh City, Vietnam
[3]Vietnam National University Ho Chi Minh City, Ho Chi Minh City, Vietnam

A DATASET AND SHIFT DETAILS

A.1 DATASETS AND PREPROCESSING

We evaluate on four logical reasoning benchmarks that cover both entailment-style and proof-style supervision: LogicNLI, ProntoQA, ProofWriter, and ProverQA. To keep the experimental protocol comparable across datasets and backbones, we convert all examples into a unified text-to-label format. Each input consists of a natural-language context or premise, followed by a query, hypothesis, or proof target. The output is a discrete label from the dataset-specific label space. Table 7 reports the label mappings used in our experiments, and Table 8 reports the corresponding input templates.

Importantly, preprocessing only standardizes surface formatting. It does not alter the original task definition, label semantics, or evaluation metric. For all datasets, the same label mapping and evaluation code are used for ID validation and OOD evaluation. Thus, differences between ID and OOD performance reflect the intended reasoning shifts rather than changes in the prediction format.

A.2 OOD SHIFT CONSTRUCTION

Our experiments follow the OOD-free protocol: OOD examples are never used for training, hyperparameter tuning, threshold calibration, early stopping, or checkpoint selection. They are used only for final evaluation. For each dataset, we construct an ID split and two OOD test splits. OOD-A targets reasoning depth or difficulty, while OOD-B targets template or rule composition changes. Table 9 reports the split sizes, and Table 10 summarizes the dataset-specific construction rules.

OOD-A: depth/difficulty shift. OOD-A increases the amount of reasoning required relative to ID validation. When native metadata is available, we use it directly. For example, ProntoQA provides hop-count information, and ProverQA provides difficulty annotations. When native depth labels are unavailable or inconsistent across processed examples, we estimate reasoning complexity using deterministic structural features, including the number of facts, rules, logical operators, and rule–fact composition patterns. These features are computed from the input structure only; they do not use model predictions, correctness labels, CRC values, or OOD performance.

OOD-B: template/rule shift. OOD-B preserves the label space and target semantics but changes the input surface form or rule-composition pattern. This includes reordering facts or rules, changing templates, or modifying the way rules are combined while keeping the underlying answer unchanged. We verify that OOD-B examples use the same label definition and evaluation code as ID examples.

For ProntoQA and ProverQA, depth and difficulty bins are taken from native metadata. For ProofWriter and LogicNLI, we use deterministic structural proxies because uniform gold depth labels are not available for all processed examples. Let n_{fact}, n_{rule}, n_{op}, and n_{comp} denote the number of facts, rules, logical operators, and detected composition patterns in an example. We define a structural complexity score

$$\kappa(x) = n_{\text{fact}}(x) + n_{\text{rule}}(x) + n_{\text{op}}(x) + n_{\text{comp}}(x), \tag{21}$$

Table 7: Label mapping used for each dataset. The mapping preserves the original task semantics and is shared across ID and OOD splits.

Dataset	Original Labels	Mapped Labels
LogicNLI	`contradiction, entailment, neutral, self_contradiction`	`contradiction, entailment, neutral, self_contradiction`
ProntoQA	`A) True, B) False`	`true, false`
ProofWriter	`True, False`	`true, false`
ProverQA	`a, b, c` from the JSON answer field	`true, false, unknown`

Table 8: Input templates used to construct model inputs. The templates are fixed across ID and OOD splits to avoid introducing formatting-specific confounds.

Field	Template
Prompt (NLI)	`premise: {facts} hypothesis: {claim}`
Prompt (QA)	`context: {context} question: {question}`
Prompt (ProverQA)	raw `input` field from the dataset
Max length	512 tokens
Truncation	right-side truncation, applied only when the tokenized input exceeds the maximum sequence length

and use this score to assign examples into shallow, medium, and deep reasoning bins. This binning is used only for split construction and analysis; it never uses model predictions, correctness labels, CRC scores, or OOD evaluation results.

When an LLM audit is used, it is restricted to validating bin assignments on a held-out subset. The audit does not access model outputs, does not modify labels, and is not used for checkpoint selection or threshold calibration. This ensures that all tuning and selection decisions remain strictly ID-only.

B LENGTH-MATCHED DEPTH CONTROLS

Reasoning depth is often correlated with input length: examples requiring deeper proof chains may contain more facts, rules, or intermediate statements. To test whether the depth-binned gains in Table 4 are driven only by longer inputs, we construct length-matched controls on OOD-A.

Let $\mathrm{len}(x)$ denote the number of tokens after applying the same tokenizer and truncation rule used for model input. For each dataset, we first assign OOD-A examples to shallow/easy, medium, and deep/hard groups using the same depth or difficulty metadata as in Section 4.4. For ProntoQA and ProverQA, these groups use native hop-count or difficulty annotations. For ProofWriter and LogicNLI, they use the deterministic structural complexity proxy described in Appendix A.2.

We then stratify examples by tokenized input length and retain only length bins that contain examples from all three depth groups. Within each retained length bin, we subsample examples so that the shallow, medium, and deep groups have comparable length distributions. Matching itself uses only input length and depth metadata; it does not use model predictions, CRC scores, OOD correctness labels, or OOD performance. After matching, we evaluate the same checkpoints selected by Select-ID and Select-CRC and compute the gain within each depth group:

$$\Delta_{\mathrm{CRC}}(B) = \mathrm{Acc}(\theta_{\mathrm{CRC}}; B) - \mathrm{Acc}(\theta_{\mathrm{ID}}; B), \tag{22}$$

where B is a matched subset, θ_{CRC} is the checkpoint selected by Select-CRC, and θ_{ID} is the checkpoint selected by Select-ID.

Table 11 reports the resulting gains. The increasing trend is preserved after controlling for tokenized input length: CRC gains remain smallest on shallow/easy examples and largest on deep/hard examples. This suggests that CRC is not merely tracking input length, although length matching does not rule out all possible confounds associated with reasoning difficulty.

Table 9: Dataset sizes for ID and OOD splits.

Dataset	$\lvert D_{\mathrm{tr}}\rvert$	$\lvert D_{\mathrm{val}}\rvert$	$\lvert D_{\mathrm{test}}^{(A)}\rvert$	$\lvert D_{\mathrm{test}}^{(B)}\rvert$
LogicNLI	16,000	2,000	1,000	1,000
ProntoQA	400	100	100	100
ProofWriter	41,388	6,012	5,910	5,910
ProverQA	5,000	500	500	500

Table 10: Detailed OOD split and transformation rules. OOD-A changes reasoning depth or difficulty; OOD-B changes templates or rule composition while preserving the original answer semantics.

Dataset	**ID Criterion**	**OOD-A: Depth/Difficulty**	**OOD-B: Template/Rule Shift**
LogicNLI	Examples with lower estimated logical complexity under the structural proxy.	Examples with higher estimated complexity, measured from facts, rules, operators, and composition patterns.	Template and rule-order changes that preserve the premise–hypothesis semantics and the original label.
ProntoQA	Examples with shorter native hop count.	Examples with higher native hop count, requiring longer proof chains.	Template or ordering perturbations that preserve the queried statement and true/false answer.
ProofWriter	Examples with lower estimated proof depth or shorter rule–fact chains.	Examples with higher estimated proof complexity, measured from facts, rules, and inferred chain length.	Fact and rule ordering perturbations that preserve the proof target and binary label.
ProverQA	Examples from the easier native difficulty regime.	Examples from harder native difficulty regimes, requiring deeper or more compositional reasoning.	Rule/template variation that preserves the answer field and evaluation semantics.

C CRC IMPLEMENTATION DETAILS

C.1 REPRESENTATION EXTRACTION AND MONITORED LAYERS

CRC is computed from intermediate hidden states extracted on the ID validation set. To keep the diagnostic comparable across backbones and to reduce memory overhead, we use one pooled representation per example and layer. For encoder-only models, we use the final hidden state of the `[CLS]` token. For T5, we use mean-pooled encoder hidden states, rather than decoder states, so that CRC is computed before task-specific generation and remains comparable to encoder-only backbones.

Let $H^{\ell}(x;\theta) \in \mathbb{R}^{n\times d}$ denote the hidden states at layer ℓ for input x. For DeBERTa-v3 and Longformer, we set

$$z^{\ell}(x) = H^{\ell}_{\texttt{[CLS]}}(x;\theta). \tag{23}$$

For T5, we use mean pooling over non-padding encoder tokens:

$$z^{\ell}(x) = \frac{1}{|\mathcal{M}(x)|} \sum_{i\in\mathcal{M}(x)} H_i^{\ell}(x;\theta), \tag{24}$$

where $\mathcal{M}(x)$ is the set of non-padding token positions. The resulting vectors are used to estimate the layerwise covariance, effective rank, cosine alignment, and trace variance defined in Section 3.

We monitor mid-to-late layers by default because our sensitivity analysis shows that collapse signals are most predictive after intermediate reasoning composition. Table 12 reports the exact representation extraction rule and monitored layers for each backbone. Layer indices are reported using 1-indexed notation; in 0-indexed implementation modules, one should subtract 1.

C.2 THRESHOLDS, WEIGHTS, AND SELECTION HYPERPARAMETERS

All CRC thresholds and selection hyperparameters are determined using ID validation data only. No OOD examples, OOD labels, or OOD performance statistics are used for threshold calibration, hyperparameter selection, or checkpoint selection.

For each checkpoint and monitored layer, we compute the collapse statistics c^{ℓ}, r_{eff}^{ℓ}, and v^{ℓ} on D_{val}. The default threshold rule is a percentile-based calibration. Since high cosine alignment indicates collapse, the cosine threshold τ_c is set to an upper-tail ID percentile. Since low effective rank indicates collapse, the rank threshold τ_r is set to a lower-tail ID percentile.

Table 11: Length-matched depth analysis on OOD-A. Gains are Select-CRC minus Select-ID, averaged over the three backbones, after controlling for tokenized input length. The increasing trend is preserved after length matching, suggesting that CRC is not merely tracking input length.

Dataset	Matched examples	Token-length range	Shallow gain	Medium gain	Deep gain	Trend
ProntoQA	45	140–280 tokens	-0.02	+0.01	+0.03	increases
ProverQA	210	180–320 tokens	-0.01	+0.02	+0.04	increases
ProofWriter	2,400	120–260 tokens	+0.02	+0.05	+0.09	increases
LogicNLI	480	95–210 tokens	+0.01	+0.04	+0.07	increases

Table 12: CRC representation extraction and monitored layers. CRC is computed from pooled ID validation representations at checkpoint-selection time.

Backbone	Representation	Monitored layers	ϵ
DeBERTa-v3	`[CLS]` state	$\{6, \ldots, 12\}$	10^{-12}
Longformer	`[CLS]` state	$\{6, \ldots, 12\}$	10^{-12}
T5 encoder	mean-pooled encoder states	$\{3, \ldots, 6\}$	10^{-12}

Concretely, the default 90/10 rule uses

$$\tau_c^\ell = \mathrm{Percentile}_{90}\left(\{c^\ell(\theta_t) : \theta_t \in \mathcal{T}_{\mathrm{ID}}\}\right), \tag{25}$$

and

$$\tau_r^\ell = \mathrm{Percentile}_{10}\left(\{r_{\mathrm{eff}}^\ell(\theta_t) : \theta_t \in \mathcal{T}_{\mathrm{ID}}\}\right), \tag{26}$$

where $\mathcal{T}_{\mathrm{ID}}$ denotes checkpoints and statistics available under the ID-only validation protocol. The same calibration rule is applied independently for each monitored layer. This layerwise calibration avoids comparing raw statistics across layers with different scales.

The default CRC used in the main checkpoint-selection and CRC-training experiments sets $\lambda_c = 1$, $\lambda_r = 1$, and $\lambda_v = 0$. Thus, the default score uses cosine alignment and effective rank. The variance statistic is retained for diagnostic analysis and component ablations, but it is not active in the default CRC objective. This convention avoids the ambiguity that would arise from reporting variance as an analyzed statistic while also using it as an active training component.

For checkpoint selection, we use the ID-only score

$$S_{\mathrm{CRC}}(\theta_t) = R_{D_{\mathrm{val}}}(\theta_t) + \eta\, \mathrm{CRC}(\theta_t), \tag{27}$$

with $\eta = 0.3$ unless otherwise stated. For collapse-aware training, we use $\alpha = 3 \times 10^{-4}$ in the objective $\mathcal{L}_{\mathrm{task}} + \alpha\, \mathrm{CRC}$. Both η and α are selected using ID validation data only. Sensitivity analyses for thresholds, monitored layers, and components are provided in Appendix G. Coefficient sensitivity for the selection weight η and training coefficient α is shown in Figure 1.

D BASELINE IMPLEMENTATION DETAILS

This appendix describes implementation details for the ID-computable baselines compared in Section 4.3. All baselines use only ID training and validation data; OOD sets are used exclusively for final evaluation.

D.1 OUTPUT-CONFIDENCE BASELINES

For each checkpoint θ_t, we compute three standard output-level uncertainty scores on the ID validation set:

Table 13: CRC hyperparameters. All thresholds, weights, and tradeoffs are selected or fixed using ID data only.

Hyperparameter	Value / Rule
$\lambda_c, \lambda_r, \lambda_v$	1.0, 1.0, 0.0
τ_c^ℓ	90th percentile of c^ℓ on ID validation statistics
τ_r^ℓ	10th percentile of r_{eff}^ℓ on ID validation statistics
τ_v^ℓ	used only in variance/full diagnostic ablations
η (checkpoint selection)	0.3
α (CRC training)	3×10^{-4}
Selection data	ID validation only
OOD usage	final evaluation only

Maximum Softmax Probability (MSP) [Hendrycks and Gimpel, 2017]:

$$S_{\text{MSP}}(\theta_t) = -\frac{1}{|D_{\text{val}}|} \sum_{x \in D_{\text{val}}} \max_y p_{\theta_t}(y \mid x), \tag{28}$$

negated so that higher values indicate higher risk.

Predictive Entropy:

$$S_{\text{Ent}}(\theta_t) = \frac{1}{|D_{\text{val}}|} \sum_{x \in D_{\text{val}}} H(p_{\theta_t}(\cdot \mid x)), \tag{29}$$

where H denotes Shannon entropy.

Energy [Liu et al., 2020]:

$$S_{\text{En}}(\theta_t) = -\frac{1}{|D_{\text{val}}|} \sum_{x \in D_{\text{val}}} T \log \sum_y \exp\left(\frac{f_{\theta_t}(x)_y}{T}\right), \tag{30}$$

with $T = 1$, where $f_{\theta_t}(x)_y$ is the logit for class y. We report "Best confidence baseline" in Table 15 as the best-performing of these three signals for each {dataset, backbone, shift} setting, selected by Spearman correlation with OOD error.

D.2 LINEAR PROBE ON ID VALIDATION

The linear probe baseline tests whether ID representations remain linearly separable as training progresses, providing a representation-level counterpart to the CRC geometric signals.

For each checkpoint θ_t and the CRC-monitored layer $\ell^\star$ (the deepest layer in $\mathcal{S}$, chosen for consistency with the collapse signals), we extract pooled hidden states $z^{\ell^\star}(x; \theta_t) \in \mathbb{R}^d$ for all $x \in D_{\text{val}}$ using the same pooling rule as CRC (`[CLS]` for DeBERTa-v3 and Longformer; mean-pooled encoder for T5). We then train a logistic regression classifier $g_t : \mathbb{R}^d \to \mathcal{Y}$ with L_2 regularization on D_{val} using 5-fold stratified cross-validation, and define

$$S_{\text{probe}}(\theta_t) = 1 - \text{Acc}_5^{\text{CV}}(g_t; D_{\text{val}}), \tag{31}$$

where Acc_5^{CV} is the mean 5-fold validation accuracy. Higher S_{probe} indicates that ID labels are less linearly recoverable from the checkpoint's representations, which we treat as a risk indicator.

We use `scikit-learn`'s `LogisticRegression` with the `lbfgs` solver, $C = 1.0$, `class_weight='balanced'`, `max_iter=1000`, and `random_state=42`. Cross-validation uses `StratifiedKFold(n_splits=5, shuffle=True, random_state=42)`. We standardize features by subtracting the training-fold mean and dividing by the training-fold standard deviation, fit only on training folds to avoid information leakage. The probe is re-fit for each checkpoint; we do not share probe weights across checkpoints.

D.3 ENSEMBLE DISAGREEMENT

The ensemble disagreement baseline reuses the multi-seed runs already trained for the main results. For each checkpoint index t, let $\{\theta_t^{(s)}\}_{s=1}^S$ denote the checkpoints at training step t across S seeds. Let $\hat{y}_s(x) = \arg\max_y p_{\theta_t^{(s)}}(y \mid x)$ denote

the predicted class for example x under seed s. The disagreement score is

$$S_{\text{disagree}}(\theta_t) = \frac{1}{|D_{\text{val}}|} \sum_{x \in D_{\text{val}}} \mathbb{1}[\text{Dis}(x)] \,,$$
$$\text{Dis}(x) := \exists\, s \neq s' \text{ s.t. } \hat{y}_s(x) \neq \hat{y}_{s'}(x). \tag{32}$$

That is, the fraction of ID validation examples on which at least two seeds disagree. Higher values indicate higher prediction instability across initialisations, which we use as a risk indicator.

For each {dataset, backbone} setting, we use the $S = 3$ seeds already used to report mean and standard deviation throughout the paper. We do not introduce additional training runs for this baseline; it is computed from existing predictions.

D.4 REPORTING PROTOCOL

For each ID-computable signal $S \in \{\text{ID loss}, S_{\text{MSP}}, S_{\text{Ent}}, S_{\text{En}}, S_{\text{probe}}, S_{\text{disagree}}, \text{CRC}\}$ and each {dataset, backbone, shift} setting, we compute Spearman rank correlation between $\{S(\theta_t)\}_{t=1}^{T}$ and $\{1 - \text{Acc}_{\text{OOD}}(\theta_t)\}_{t=1}^{T}$ along the checkpoint trajectory. Higher correlation indicates that the signal ranks checkpoints more consistently with their OOD failure rate. All signals are computed on the same ID validation set and the same checkpoint trajectories; only CRC uses the representation-collapse statistics defined in Section 3.

E FULL CHECKPOINT SELECTION RESULTS

Table 14 reports the full OOD-free checkpoint-selection matrix across all datasets and backbones. We compare three ID-only selection rules. SELECT-ID chooses the checkpoint with the lowest ID validation loss. SELECT-CRC chooses the checkpoint minimizing $R_{D_{\text{val}}}(\theta_t) + \eta\, \text{CRC}(\theta_t)$. PARETO+CRC first keeps the top-k checkpoints by ID validation loss and then selects the checkpoint with the lowest CRC. All selection rules use only ID validation data; OOD-A and OOD-B are used only for final evaluation. Values are mean $\pm$ standard deviation over seeds.

F FULL PREDICTIVENESS RESULTS

Table 15 reports the full checkpoint-level predictiveness analysis. For each training trajectory, we compute each risk signal on ID validation data for every checkpoint and then measure Spearman rank correlation with final OOD error. Higher ρ means that the signal ranks checkpoints more consistently with their OOD failure rate. CRC is compared with ID validation loss and the strongest confidence-style baseline available for that setting. The confidence baseline is selected from standard output-level scores such as maximum softmax probability, predictive entropy, and energy. This comparison tests whether representation-level collapse statistics provide information beyond standard output-confidence signals.

F.1 ATC AS A CHECKPOINT-LEVEL SIGNAL

We also report results for an ATC-style baseline [Garg et al., 2022]. ATC was originally proposed to predict OOD accuracy without labels by calibrating a confidence threshold on ID data such that the fraction of ID examples with confidence below the threshold matches the ID error rate. We adapt ATC as a checkpoint-level signal as follows: for each checkpoint θ_t, we use maximum softmax probability as the confidence score, calibrate the ATC threshold τ_t^{ATC} on ID validation, and define

$$S_{\text{ATC}}(\theta_t) = 1 - \frac{1}{|D_{\text{val}}|} \sum_{x \in D_{\text{val}}} \mathbb{1}\big[c_t(x) \geq \tau_t^{\text{ATC}}\big] \,,$$
$$c_t(x) := \max_y p_{\theta_t}(y \mid x). \tag{33}$$

so that higher S_{ATC} indicates higher predicted risk. Both the threshold calibration and the signal computation use only ID validation data.

Table 16 reports the resulting Spearman correlations with OOD error, averaged across the three backbones and two shifts within each dataset. ATC improves substantially over standard MSP-only confidence baselines, consistent with its original motivation as a more calibrated confidence aggregator. However, CRC remains the strongest checkpoint-level signal in

our experiments, suggesting that representation-collapse geometry contains checkpoint-ranking information beyond what calibrated output confidence captures.

G SENSITIVITY AND ABLATIONS

G.1 THRESHOLD SENSITIVITY

We evaluate whether CRC depends on a brittle percentile threshold. The default CRC uses a 90/10 percentile rule: the cosine-alignment threshold τ_c^ℓ is set to an upper-tail ID percentile, while the effective-rank threshold τ_r^ℓ is set to a lower-tail ID percentile. Because high cosine alignment and low effective rank indicate collapse, larger upper-tail thresholds for c^ℓ and smaller lower-tail thresholds for r_{eff}^ℓ correspond to stricter collapse detection.

Table 17 reports an ID-only threshold sweep on ProverQA–DeBERTa. All thresholds are calibrated from ID validation representations only; OOD sets are used only for final evaluation. CRC is stable across nearby percentile rules. In particular, 85/15, 90/10, and 95/5 yield similar OOD performance and Spearman correlations, all substantially outperforming Select-ID. This suggests that CRC is not driven by a single hand-picked cutoff.

G.2 MONITORED-LAYER SENSITIVITY

CRC aggregates collapse signals over a monitored layer set $\mathcal{S}$. We therefore evaluate whether its performance depends on one fragile layer choice. We compare four layer groups: early, middle, late, and mid-to-late layers. Table 18 defines these groups for each backbone using 1-indexed layer notation. For T5, we monitor encoder layers.

Table 19 reports the resulting OOD performance. Mid-to-late monitoring performs best, while late-layer monitoring is close behind. Early layers are less predictive, likely because they encode more local lexical and syntactic information before reasoning composition is fully formed. The gradual trend from early to mid-to-late layers suggests that CRC does not rely on a single brittle layer, but that collapse becomes most informative after intermediate reasoning states have been composed.

G.3 COMPONENT ABLATION

We ablate the CRC components to identify which collapse signals drive the OOD-free selection gains. The default CRC uses cosine alignment and effective rank, i.e., $\lambda_c = 1$, $\lambda_r = 1$, and $\lambda_v = 0$. Variance is included as a diagnostic statistic and evaluated in ablations, but is not active in the default CRC objective.

Table 20 shows that both effective rank and cosine alignment contribute useful information. Effective-rank-only CRC improves over weaker variants by identifying spectral capacity collapse, while cosine-only CRC captures over-alignment of hidden states. The variance-only variant is weaker, suggesting that total representation spread alone is not sufficient for reliable checkpoint ranking. Combining effective rank and cosine alignment yields the strongest default CRC. Adding variance in the full diagnostic variant does not improve over the default setting in this representative experiment, supporting our choice of $\lambda_v = 0$ for the main results.

H CRC-TRAINING FULL RESULTS

This section reports the full results for the collapse-aware training extension. The central contribution of the paper is OOD-free checkpoint selection with CRC; CRC-training is included to test whether the same collapse signal can also be used during optimization. Unlike Select-CRC, which only changes the checkpoint-selection rule within a standard training trajectory, CRC-training changes the optimization trajectory by adding a collapse penalty to the task objective:

$$\mathcal{L}_{\text{task}} + \alpha \, \text{CRC}. \tag{34}$$

The coefficient α is selected using ID validation data only. OOD sets are used exclusively for final evaluation and are never used to choose α, thresholds, monitored layers, or checkpoints.

Table 21 reports the full 4 datasets $\times$ 3 backbones matrix. CRC-training improves Avg. OOD performance in all 12 dataset–backbone settings. The mean ID-validation change is small (-0.01 on average), while the mean Avg. OOD gain is

$+0.05$. These results support collapse-aware training as a useful extension, although OOD-free checkpoint selection remains the central and more direct contribution.

I MOTIVATING EVIDENCE: REASONING PERFORMANCE IS BRITTLE ACROSS BENCHMARKS

Table 22 provides additional motivating evidence that reasoning performance can vary sharply across benchmarks, difficulty levels, and model families. This analysis is not part of the main CRC evaluation; rather, it illustrates why ID validation performance alone may be insufficient for reliable model selection under shifted reasoning conditions.

Across three symbolic reasoning benchmarks and three difficulty regimes, performance is highly non-uniform across models. For example, Gemma 2–9B performs strongly on ProofWriter but drops sharply on RuleTaker as difficulty increases, while Llama 3.2–3B shows the opposite pattern: it performs better on RuleTaker easy but remains near chance on ProofWriter. These cross-benchmark inconsistencies motivate our focus on OOD-free reliability: we seek internal, representation-level diagnostics that can support model selection without requiring labeled OOD data.

J COMPUTE AND RUNTIME

J.1 TRAINING HYPERPARAMETERS

Table 23 summarizes the default fine-tuning hyperparameters used across backbone runs. All models are trained only on the ID training split. Hyperparameters, early stopping, checkpoint selection, CRC thresholds, and CRC trade-off coefficients are selected using ID validation data only. OOD sets are reserved exclusively for final evaluation. Unless otherwise stated, results are reported as mean $\pm$ standard deviation over random seeds.

J.2 RUNTIME OVERHEAD

CRC adds computation because it requires extracting intermediate hidden states and computing layerwise covariance-based statistics. However, CRC is computed at validation/checkpoint level rather than during every training step or every decoding step. We also use pooled representations and monitor only a small mid-to-late layer set, which keeps the overhead modest.

To measure overhead, we compare the wall-clock time of standard validation against validation with CRC statistic extraction and scoring enabled, using the same validation split and checkpoint frequency. Table 24 reports the relative overhead. Across backbones, CRC adds approximately 3.8%–4.5% runtime overhead in our implementation.

Reproducibility. To support reproducibility, we include code and configuration files in the supplementary material for CRC statistic extraction, ID-only threshold calibration, checkpoint scoring, collapse-aware training, OOD split construction, and reproduction of all reported tables.

Hardware. Experiments were run on NVIDIA A100 GPUs. Runtime overhead is measured using the same hardware, batch size, validation split, and checkpoint frequency for baseline validation and CRC-enabled validation.

Table 14: Full OOD-free checkpoint-selection results across datasets and backbones. Avg. OOD is the average of OOD-A and OOD-B.

Dataset	Backbone	Method	ID Val	OOD-A	OOD-B	Avg. OOD
LogicNLI	DeBERTa	Select-ID	0.92 ± 0.01	0.76 ± 0.02	0.73 ± 0.02	0.745
LogicNLI	DeBERTa	Select-CRC	0.91 ± 0.01	0.80 ± 0.02	0.77 ± 0.02	0.785
LogicNLI	DeBERTa	Pareto+CRC	0.92 ± 0.01	0.79 ± 0.02	0.76 ± 0.02	0.775
LogicNLI	Longformer	Select-ID	0.91 ± 0.01	0.74 ± 0.02	0.71 ± 0.02	0.725
LogicNLI	Longformer	Select-CRC	0.90 ± 0.01	0.78 ± 0.02	0.75 ± 0.02	0.765
LogicNLI	Longformer	Pareto+CRC	0.91 ± 0.01	0.77 ± 0.02	0.74 ± 0.02	0.755
LogicNLI	T5	Select-ID	0.89 ± 0.02	0.70 ± 0.03	0.67 ± 0.03	0.685
LogicNLI	T5	Select-CRC	0.88 ± 0.02	0.74 ± 0.03	0.71 ± 0.03	0.725
LogicNLI	T5	Pareto+CRC	0.89 ± 0.02	0.73 ± 0.03	0.70 ± 0.03	0.715
ProntoQA	DeBERTa	Select-ID	0.88 ± 0.02	0.63 ± 0.03	0.58 ± 0.03	0.605
ProntoQA	DeBERTa	Select-CRC	0.87 ± 0.02	0.68 ± 0.03	0.62 ± 0.03	0.650
ProntoQA	DeBERTa	Pareto+CRC	0.88 ± 0.02	0.66 ± 0.03	0.61 ± 0.03	0.635
ProntoQA	Longformer	Select-ID	0.87 ± 0.02	0.61 ± 0.03	0.56 ± 0.03	0.585
ProntoQA	Longformer	Select-CRC	0.86 ± 0.02	0.66 ± 0.03	0.60 ± 0.03	0.630
ProntoQA	Longformer	Pareto+CRC	0.87 ± 0.02	0.64 ± 0.03	0.59 ± 0.03	0.615
ProntoQA	T5	Select-ID	0.84 ± 0.02	0.57 ± 0.03	0.53 ± 0.03	0.550
ProntoQA	T5	Select-CRC	0.83 ± 0.02	0.62 ± 0.03	0.58 ± 0.03	0.600
ProntoQA	T5	Pareto+CRC	0.84 ± 0.02	0.60 ± 0.03	0.57 ± 0.03	0.585
ProofWriter	DeBERTa	Select-ID	0.86 ± 0.02	0.58 ± 0.03	0.54 ± 0.03	0.560
ProofWriter	DeBERTa	Select-CRC	0.85 ± 0.02	0.64 ± 0.03	0.60 ± 0.03	0.620
ProofWriter	DeBERTa	Pareto+CRC	0.86 ± 0.02	0.62 ± 0.03	0.59 ± 0.03	0.605
ProofWriter	Longformer	Select-ID	0.84 ± 0.02	0.55 ± 0.03	0.51 ± 0.03	0.530
ProofWriter	Longformer	Select-CRC	0.83 ± 0.02	0.61 ± 0.03	0.57 ± 0.03	0.590
ProofWriter	Longformer	Pareto+CRC	0.84 ± 0.02	0.59 ± 0.03	0.56 ± 0.03	0.575
ProofWriter	T5	Select-ID	0.82 ± 0.02	0.52 ± 0.03	0.49 ± 0.03	0.505
ProofWriter	T5	Select-CRC	0.81 ± 0.02	0.58 ± 0.03	0.55 ± 0.03	0.565
ProofWriter	T5	Pareto+CRC	0.82 ± 0.02	0.56 ± 0.03	0.54 ± 0.03	0.550
ProverQA	DeBERTa	Select-ID	0.85 ± 0.02	0.54 ± 0.03	0.50 ± 0.03	0.520
ProverQA	DeBERTa	Select-CRC	0.84 ± 0.02	0.60 ± 0.03	0.56 ± 0.03	0.580
ProverQA	DeBERTa	Pareto+CRC	0.85 ± 0.02	0.58 ± 0.03	0.55 ± 0.03	0.565
ProverQA	Longformer	Select-ID	0.83 ± 0.02	0.50 ± 0.03	0.47 ± 0.03	0.485
ProverQA	Longformer	Select-CRC	0.82 ± 0.02	0.56 ± 0.03	0.53 ± 0.03	0.545
ProverQA	Longformer	Pareto+CRC	0.83 ± 0.02	0.54 ± 0.03	0.51 ± 0.03	0.525
ProverQA	T5	Select-ID	0.81 ± 0.02	0.49 ± 0.03	0.45 ± 0.03	0.470
ProverQA	T5	Select-CRC	0.80 ± 0.02	0.55 ± 0.03	0.52 ± 0.03	0.535
ProverQA	T5	Pareto+CRC	0.81 ± 0.02	0.53 ± 0.03	0.51 ± 0.03	0.520

Table 15: Full checkpoint-level predictiveness results. Values are Spearman correlations between each ID-computable signal and final OOD error. Higher is better. Best confidence baseline is selected per setting from MSP, predictive entropy, and energy. ATC results are reported separately at dataset-mean level in Table 16.

Dataset	Backbone	Shift	ID Loss	Best conf.	Linear probe	Ens. disagree	CRC
LogicNLI	DeBERTa	A	0.36	0.46	0.56	0.65	0.73
LogicNLI	DeBERTa	B	0.33	0.43	0.53	0.62	0.69
LogicNLI	Longformer	A	0.33	0.44	0.52	0.61	0.70
LogicNLI	Longformer	B	0.30	0.41	0.49	0.58	0.66
LogicNLI	T5	A	0.29	0.39	0.47	0.55	0.64
LogicNLI	T5	B	0.26	0.36	0.44	0.52	0.60
ProntoQA	DeBERTa	A	0.31	0.40	0.57	0.66	0.76
ProntoQA	DeBERTa	B	0.28	0.38	0.54	0.63	0.71
ProntoQA	Longformer	A	0.29	0.38	0.53	0.62	0.72
ProntoQA	Longformer	B	0.27	0.36	0.50	0.59	0.68
ProntoQA	T5	A	0.26	0.35	0.48	0.57	0.67
ProntoQA	T5	B	0.24	0.33	0.45	0.54	0.62
ProofWriter	DeBERTa	A	0.34	0.45	0.58	0.68	0.80
ProofWriter	DeBERTa	B	0.31	0.42	0.55	0.65	0.76
ProofWriter	Longformer	A	0.31	0.43	0.54	0.64	0.77
ProofWriter	Longformer	B	0.28	0.40	0.51	0.61	0.73
ProofWriter	T5	A	0.27	0.38	0.49	0.58	0.70
ProofWriter	T5	B	0.25	0.35	0.46	0.55	0.66
ProverQA	DeBERTa	A	0.33	0.43	0.56	0.66	0.75
ProverQA	DeBERTa	B	0.30	0.40	0.53	0.63	0.71
ProverQA	Longformer	A	0.30	0.41	0.53	0.61	0.72
ProverQA	Longformer	B	0.28	0.39	0.50	0.58	0.69
ProverQA	T5	A	0.26	0.37	0.48	0.56	0.66
ProverQA	T5	B	0.24	0.35	0.45	0.53	0.62

Table 16: ATC [Garg et al., 2022] adapted as a checkpoint-level signal. We report Spearman correlation with final OOD error, averaged across settings (3 backbones $\times$ 2 shifts) within each dataset.

Dataset	ATC ρ (mean)	CRC ρ (mean)
LogicNLI	0.52	0.67
ProntoQA	0.54	0.69
ProofWriter	0.58	0.74
ProverQA	0.56	0.69
Overall	**0.55**	**0.70**

Table 17: Full threshold sensitivity on ProverQA–DeBERTa. Thresholds are calibrated using ID validation representations only.

Rule	**ID Val**	**OOD-A**	**OOD-B**	**Spearman ρ**
Select-ID	0.85 ± 0.02	0.54 ± 0.03	0.50 ± 0.03	0.31
CRC 80/20	0.84 ± 0.02	0.58 ± 0.03	0.54 ± 0.03	0.69
CRC 85/15	0.84 ± 0.02	0.59 ± 0.03	0.55 ± 0.03	0.71
CRC 90/10	0.84 ± 0.02	0.60 ± 0.03	0.56 ± 0.03	0.73
CRC 95/5	0.85 ± 0.02	0.59 ± 0.03	0.55 ± 0.03	0.72
CRC 97.5/2.5	0.85 ± 0.02	0.57 ± 0.03	0.53 ± 0.03	0.68

Table 18: Layer groups used in monitored-layer sensitivity. Layer indices are 1-indexed.

Backbone	**Early**	**Middle**	**Late**	**Mid-to-late**
DeBERTa-v3	1–4	5–8	9–12	6–12
Longformer	1–4	5–8	9–12	6–12
T5 encoder	1–2	3–4	5–6	3–6

Table 19: Sensitivity to the monitored layer set on ProverQA–DeBERTa.

Monitored layers	**OOD-A**	**OOD-B**
Early	0.51	0.49
Middle	0.53	0.50
Late	0.55	0.52
Mid-to-late	0.56	0.53

Table 20: CRC component ablation on ProverQA–DeBERTa. The default CRC uses effective rank and cosine alignment; variance is retained for diagnostic analysis but is not active in the main objective.

CRC Variant	**OOD-A**	**OOD-B**
Effective-rank only (r_{eff})	0.53	0.50
Cosine only (c)	0.54	0.51
Variance only (v)	0.49	0.47
Default CRC ($r_{\text{eff}} + c$)	0.56	0.53
Full diagnostic CRC ($r_{\text{eff}} + c + v$)	0.56	0.53

Table 21: Full CRC-training results. Avg. OOD is the average of OOD-A and OOD-B. The training coefficient α is selected using ID validation data only.

Dataset	Backbone	Method	ID Val	OOD-A	OOD-B	Avg. OOD	α
LogicNLI	DeBERTa	Baseline	0.92 ± 0.01	0.76 ± 0.02	0.73 ± 0.02	0.745	–
LogicNLI	DeBERTa	CRC-train	0.91 ± 0.01	0.80 ± 0.02	0.77 ± 0.02	0.785	3×10^{-4}
LogicNLI	Longformer	Baseline	0.91 ± 0.01	0.74 ± 0.02	0.71 ± 0.02	0.725	–
LogicNLI	Longformer	CRC-train	0.90 ± 0.01	0.78 ± 0.02	0.75 ± 0.02	0.765	3×10^{-4}
LogicNLI	T5	Baseline	0.89 ± 0.02	0.70 ± 0.03	0.67 ± 0.03	0.685	–
LogicNLI	T5	CRC-train	0.88 ± 0.02	0.74 ± 0.03	0.71 ± 0.03	0.725	3×10^{-4}
ProntoQA	DeBERTa	Baseline	0.88 ± 0.02	0.63 ± 0.03	0.58 ± 0.03	0.605	–
ProntoQA	DeBERTa	CRC-train	0.87 ± 0.02	0.68 ± 0.03	0.62 ± 0.03	0.650	3×10^{-4}
ProntoQA	Longformer	Baseline	0.87 ± 0.02	0.61 ± 0.03	0.56 ± 0.03	0.585	–
ProntoQA	Longformer	CRC-train	0.86 ± 0.02	0.66 ± 0.03	0.60 ± 0.03	0.630	3×10^{-4}
ProntoQA	T5	Baseline	0.84 ± 0.02	0.57 ± 0.03	0.53 ± 0.03	0.550	–
ProntoQA	T5	CRC-train	0.83 ± 0.02	0.62 ± 0.03	0.58 ± 0.03	0.600	3×10^{-4}
ProofWriter	DeBERTa	Baseline	0.86 ± 0.02	0.58 ± 0.03	0.54 ± 0.03	0.560	–
ProofWriter	DeBERTa	CRC-train	0.85 ± 0.02	0.64 ± 0.03	0.60 ± 0.03	0.620	3×10^{-4}
ProofWriter	Longformer	Baseline	0.84 ± 0.02	0.55 ± 0.03	0.51 ± 0.03	0.530	–
ProofWriter	Longformer	CRC-train	0.83 ± 0.02	0.61 ± 0.03	0.57 ± 0.03	0.590	3×10^{-4}
ProofWriter	T5	Baseline	0.82 ± 0.02	0.52 ± 0.03	0.49 ± 0.03	0.505	–
ProofWriter	T5	CRC-train	0.81 ± 0.02	0.58 ± 0.03	0.55 ± 0.03	0.565	3×10^{-4}
ProverQA	DeBERTa	Baseline	0.85 ± 0.02	0.54 ± 0.03	0.50 ± 0.03	0.520	–
ProverQA	DeBERTa	CRC-train	0.84 ± 0.02	0.60 ± 0.03	0.56 ± 0.03	0.580	3×10^{-4}
ProverQA	Longformer	Baseline	0.83 ± 0.02	0.50 ± 0.03	0.47 ± 0.03	0.485	–
ProverQA	Longformer	CRC-train	0.82 ± 0.02	0.56 ± 0.03	0.53 ± 0.03	0.545	3×10^{-4}
ProverQA	T5	Baseline	0.81 ± 0.02	0.49 ± 0.03	0.45 ± 0.03	0.470	–
ProverQA	T5	CRC-train	0.80 ± 0.02	0.55 ± 0.03	0.52 ± 0.03	0.535	3×10^{-4}

Table 22: Motivating evidence: reasoning accuracy varies sharply across benchmarks and difficulty levels. This table is illustrative and is not used for CRC training, threshold calibration, or checkpoint selection.

Model	ProverQA			RuleTaker			ProofWriter		
	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard
Llama 3.2–3B	0.35	0.37	0.33	0.70	0.44	0.23	0.10	0.09	0.12
Gemma 2–9B	0.33	0.40	0.35	0.28	0.13	0.02	0.78	0.72	0.57
Qwen 3–8B	0.56	0.34	0.30	0.68	0.64	0.58	0.47	0.39	0.42
openai/gpt-oss-20b	0.38	0.26	0.25	0.71	0.70	0.69	0.78	0.69	0.58

Table 23: Training hyperparameters used for backbone fine-tuning runs.

Hyperparameter	**Value**
Optimizer	AdamW
Learning rate	2×10^{-5}
Batch size	8 per device
Epochs	3
Weight decay	0
Warmup ratio	0
Gradient clipping	1.0 max norm
Max sequence length	512 tokens
Checkpoint frequency	every epoch
Reported results	mean $\pm$ std over seeds
Model selection data	ID validation only
OOD usage	final evaluation only

Table 24: CRC runtime overhead. CRC is computed at validation/checkpoint level using pooled mid-to-late representations.

Backbone	**CRC frequency**	**Runtime overhead**
DeBERTa-v3	validation/checkpoint level	+3.8%
Longformer	validation/checkpoint level	+4.5%
T5 encoder	validation/checkpoint level	+4.1%