BASICLT: Basic-Level Abstraction and Selective Differentiation for Long-Tailed Recognition

Yuan Dong[1] **Di Wu**[1] **Zhe Zhao**[1] **Liheng Yu**[1]
Xiaofeng Cao[3] **Pengkun Wang**[1,2,*] **Yang Wang**[1,2,*]
[1]University of Science and Technology of China
[2]Suzhou Institute for Advanced Research
[3]Tongji University
[*]Corresponding authors

Abstract

Long-tailed recognition is commonly formulated as a problem of imbalanced supervision, where rare classes suffer from insufficient training examples. In this work, we study a complementary source of difficulty: under severe scarcity, tail classes may be forced into fine-grained discrimination before the model has acquired sufficiently reliable coarse semantic structure. We refer to this phenomenon as *granularity mismatch under scarcity*. To address it, we propose BASICLT, a two-stage framework that augments an underlying fine-grained recognition model with basic-level abstraction and commitment-guided selective differentiation. In Stage 1, BASICLT learns an explicit basic-level branch and aligns it with the family-level distribution induced by the fine-grained predictor, encouraging stable coarse semantic organization before additional fine-grained refinement is introduced. In Stage 2, a frozen Stage 1 teacher provides stable family-level commitment signals, and a family-conditioned residual refinement module is trained only on committed medium-shot and few-shot samples. At inference time, the residual correction is activated conservatively according to student-side commitment, so that fine-grained refinement acts as a conditional within-family correction rather than an unconditional auxiliary classifier. We evaluate BASICLT on CIFAR-100-LT, CIFAR-10-LT, ImageNet-LT, iNaturalist 2018, and Places-LT. Across these benchmarks, BASICLT achieves competitive performance against strong long-tailed baselines, with the most consistent gains appearing in medium-shot and few-shot regimes. Ablations and analyses further show that both basic-level abstraction and commitment-guided selective refinement are important, supporting the view that controlling the semantic granularity of discrimination can complement conventional rebalancing strategies for long-tailed learning. Code is available at **Supplement**.

1 Introduction

Long-tailed recognition remains a central challenge in visual learning because real-world category distributions are highly imbalanced. A small number of head classes provide abundant supervision, while many medium-shot and few-shot classes are observed only sparsely. As a result, standard deep classifiers tend to favor frequent categories and often fail to learn reliable decision boundaries for rare classes. A large body of work has addressed this problem through re-sampling, loss re-weighting, logit adjustment, decoupled training, contrastive learning, and expert-based specialization [1–6]. These methods have substantially improved long-tailed recognition. However, most of them still impose the final recognition objective in a flat fine-grained label space, where all classes are encouraged to compete at the same semantic resolution throughout training.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

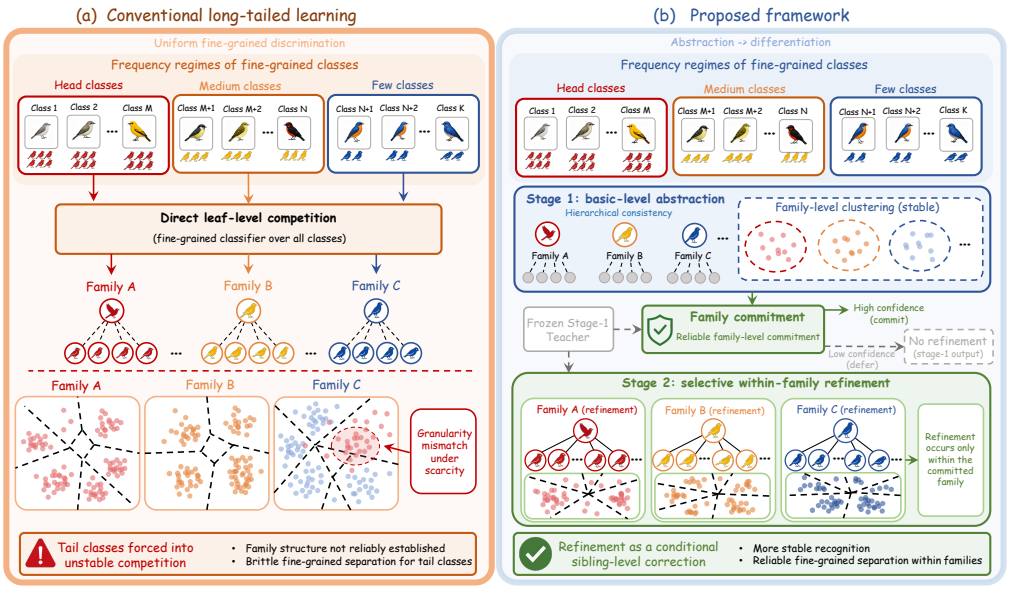


Figure 1: Conceptual comparison between conventional long-tailed learning and BASICLT. Conventional long-tailed training applies flat fine-grained discrimination to all classes from the outset, which can force scarce classes into unstable sibling-level competition. BASICLT follows an abstraction-to-differentiation progression: it first learns stable basic-level structure, estimates family-level commitment, and then selectively applies within-family residual refinement only when the commitment is reliable.

In this work, we study a complementary difficulty arising under severe data scarcity. For rare classes, the available supervision may be insufficient to support reliable fine-grained separation from visually similar sibling classes, even though identifying the correct coarse semantic family may already be feasible. In such cases, the learning objective can demand leaf-level discrimination before the model has acquired stable family-level structure. We refer to this discrepancy as *granularity mismatch under scarcity*: the semantic granularity imposed by the objective may be too fine relative to the evidence available for low-frequency classes. This perspective does not replace the conventional imbalance view; rather, it suggests that long-tailed recognition may depend not only on how supervision is rebalanced, but also on when and at what semantic granularity discrimination is introduced.

This observation motivates a staged allocation of additional discriminative capacity. Stage 1 retains fine-grained supervision while learning an explicit family scaffold and coarse–fine consistency. Stage 2 then introduces extra sibling-level residual capacity only when family commitment is reliable. The working principle is therefore that *basic-level abstraction precedes additional selective differentiation, while fine-grained learning is preserved*. It does not require coarse-only pretraining or postponing the target fine-grained objective.

Based on this principle, we propose BASICLT, a two-stage framework for long-tailed recognition. The method assumes a fixed basic-level taxonomy that maps each fine-grained class to a coarse semantic family. This taxonomy is used not to change the final task, which remains fine-grained recognition, but to provide an intermediate semantic scaffold for training and selective refinement.

In Stage 1, BASICLT augments an underlying fine-grained recognition model with a basic-level prediction branch and a hierarchical consistency objective. The basic branch learns family-level supervision, while the consistency objective aligns the family-level distribution induced by the fine-grained predictor with explicit basic-level prediction. This stage encourages the shared representation to acquire stable coarse semantic organization before additional fine-grained refinement is introduced.

In Stage 2, BASICLT introduces a family-conditioned residual refinement module for sibling-level differentiation. Crucially, this refinement is not applied uniformly. A frozen Stage 1 teacher provides stable family-level commitment signals, and refinement is trained primarily on committed medium-shot and few-shot samples. The residual branch is regularized and used as a conditional correction on

the base predictor, rather than as an unconstrained auxiliary classifier. During inference, the student activates the residual correction only when its own family-level commitment is sufficiently reliable. Thus, the framework separates basic-level abstraction from sibling-level differentiation and controls the semantic granularity of discrimination according to confidence in the learned coarse structure.

We evaluate BASICLT on five standard long-tailed benchmarks: CIFAR-100-LT, CIFAR-10-LT, ImageNet-LT, iNaturalist 2018, and Places-LT. Across these datasets, BASICLT achieves competitive performance against strong baselines, with gains most pronounced in medium- and few-shot regimes where premature fine-grained competition is most fragile. Beyond overall accuracy, our analyses further reveal that these improvements stem from both framework components: Stage 1 enhances basic-level abstraction and coarse–fine consistency, while Stage 2 utilizes commitment-guided selective refinement instead of uniform residual correction.

In summary, our contributions are as follows:

❶ **New perspective.** We identify *granularity mismatch under scarcity* as a critical challenge in long-tailed recognition, highlighting that forcing fine-grained discrimination on rare classes is premature before establishing a stable coarse semantic structure.

❷ **Framework design.** We propose BASICLT, a two-stage abstraction-to-differentiation framework that sequences basic-level structure learning with teacher-stabilized, commitment-guided residual refinement to handle low-frequency category differentiation.

❸ **Empirical evidence.** We evaluate BASICLT on CIFAR-100-LT, CIFAR-10-LT, ImageNet-LT, iNaturalist 2018, and Places-LT, and provide controlled analyses showing that abstraction, hierarchical consistency, and selective refinement collectively drive the final performance.

2 Related Work

Long-tailed recognition. Long-tailed recognition has been studied via data rebalancing, loss correction, decoupled training, and specialized architectures [1, 4, 3, 7, 5, 8, 9]. Prototype-guided manifold completion further improves tail representations by recovering underrepresented semantic structures [10]. Related studies address data imbalance and representational interference through cross-silo feature alignment [11], non-interfering attention masking for federated prompt refinement [12], and geometry-aware exemplar replay under continual dynamic heterogeneity [13]. While these approaches improve learning under imbalance or heterogeneity, their primary focus is not on adapting semantic resolution to class scarcity. In contrast, we vary semantic granularity under scarcity, leveraging family-level abstraction as a gating mechanism for fine-grained refinement rather than just an auxiliary signal.

Hierarchy-aware visual recognition. Taxonomies and coarse-to-fine structure have long supported visual recognition. HD-CNN [14], CNN-RNN [15], and hierarchy-aware supervision [16] exploit relations among category levels. For long-tailed recognition, SuperDisco [17] discovers superclass graph representations and rectifies image features through message passing. BASICLT addresses a complementary question: given a fixed, single-membership taxonomy, when should additional sibling-level discrimination be learned and activated? Its contribution is not superclass supervision itself, but the combination of learned family abstraction, frozen-teacher selection, constrained residual differentiation, and student-side commitment with fallback. The flat fine-grained predictor is retained throughout. Taxonomy construction and matched controls are detailed in Appendices C and H.

Basic-level abstraction and category learning. Research in cognitive science suggests that human categorization often operates at an intermediate, or basic, level of abstraction [18]. More recent evidence indicates that visual representations may also prioritize this level during recognition [19]. Our work is inspired by this perspective, but does not aim to model human categorization directly. Instead, we use basic-level abstraction as a training principle for long-tailed recognition and combine it with commitment-guided selective refinement. More detailed discussion is provided in Appendix A.

3 Method

We propose BASICLT, a two-stage framework for long-tailed recognition based on an abstraction-to-differentiation progression. The framework is motivated by the observation that, under severe

data scarcity, fine-grained decision boundaries may be unreliable before the model has acquired stable coarse semantic structure. Instead of imposing additional fine-grained discrimination uniformly across all classes, BASICLT first learns a basic-level abstraction layer and then introduces sibling-level differentiation only when the corresponding family-level prediction is sufficiently reliable.

The final task remains standard fine-grained recognition over the original label space. The basic-level taxonomy is not used to replace the fine-grained labels or to change the evaluation protocol. Rather, it serves as an intermediate semantic scaffold. Stage 1 uses this scaffold to learn explicit family-level abstraction and to align the fine-grained predictor with the induced family-level structure. Stage 2 then trains a family-conditioned residual branch for within-family refinement, with both training and inference controlled by family-level commitment.

3.1 Problem Setup

We consider long-tailed recognition over a fine-grained label space $\mathcal{Y} = \{1, \dots, K\}$ with training set

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}, \tag{1}$$

where $x_i \in \mathcal{X}$ is an input image and $y_i \in \mathcal{Y}$ is its fine-grained class label. The class distribution is imbalanced: a small number of head classes contain abundant training examples, while many medium-shot and few-shot classes are observed only sparsely.

In addition to the fine-grained label space, we assume a fixed basic-level label space $\mathcal{B} = \{1, \dots, M\}$ and a deterministic mapping

$$g : \mathcal{Y} \to \mathcal{B}, \qquad b_i = g(y_i), \tag{2}$$

which assigns each fine-grained class to exactly one basic-level family. The mapping g is fixed before training and is not optimized jointly with the model. It is used only as an intermediate semantic scaffold; all final evaluations are performed on the original fine-grained label space $\mathcal{Y}$.

Let f_θ denote the visual encoder and let

$$z = f_\theta(x) \tag{3}$$

be the learned representation of an input image x. The fine-grained predictor produces logits

$$s(x) \in \mathbb{R}^K, \tag{4}$$

which are used for the original recognition task. On top of the same representation, BASICLT introduces a basic-level branch that produces family logits

$$u(x) \in \mathbb{R}^M. \tag{5}$$

Thus, the model maintains two views of the same input: $s(x)$ describes leaf-level discrimination, while $u(x)$ describes basic-level abstraction.

To compare these two prediction spaces, we define an aggregation operator $\mathrm{Agg} : \mathbb{R}^K \to \mathbb{R}^M$ that maps fine-grained logits to family-level logits:

$$\mathrm{Agg}(s(x))_b = \log \sum_{y : g(y) = b} \exp\big(s_y(x)\big), \qquad b \in \mathcal{B}. \tag{6}$$

This aggregation has a probability-level interpretation:

$$\mathrm{softmax}(\mathrm{Agg}(s(x)))_b = \sum_{y : g(y) = b} \mathrm{softmax}(s(x))_y. \tag{7}$$

Thus, $\mathrm{softmax}(\mathrm{Agg}(s(x)))$ is the family posterior conditioned on x, obtained by summing the fine-class probabilities within each family; $\mathrm{Agg}(s(x))$ itself contains family logits. This places both predictors at the same semantic level. Appendix B.1 clarifies the temperature convention used for consistency.

Following standard long-tailed protocols, we partition fine-grained classes into many-shot, medium-shot, and few-shot groups according to their training frequency. Let n_y denote the number of training examples of class y, and let

$$\gamma(y) \in \{\mathrm{many}, \mathrm{medium}, \mathrm{few}\} \tag{8}$$

denote its frequency regime. We further define the set of lower-frequency classes as

$$\mathcal{Y}_{\mathrm{mf}} = \{y \in \mathcal{Y} \mid \gamma(y) \in \{\mathrm{medium}, \mathrm{few}\}\}. \tag{9}$$

Stage 2 uses this set to focus residual refinement on the regimes where additional fine-grained differentiation is most needed.

3.2 Stage 1: Basic-Level Abstraction

The goal of Stage 1 is to establish reliable family-level structure before introducing additional fine-grained refinement. Under long-tailed scarcity, medium-shot and few-shot classes may not provide enough evidence for stable sibling-level decision boundaries early in training. However, related fine-grained classes can still share useful structure at the basic level. Stage 1 therefore augments the fine-grained predictor with an explicit basic-level branch and encourages consistency between the supervised family prediction and the family-level distribution induced by the fine-grained logits.

The first abstraction objective is direct basic-level supervision:

$$\mathcal{L}_{\text{basic}} = \frac{1}{N}\sum_{i=1}^{N} \text{CE}\left(u(x_i), b_i\right), \tag{10}$$

where CE denotes cross-entropy. This term encourages the shared representation to capture semantic regularities that are common to fine-grained classes within the same family.

The second abstraction objective is hierarchical consistency. Since the basic-level branch is directly supervised by family labels, we use it as a stable abstraction target and align the family-level distribution induced by the fine-grained predictor to it:

$$\mathcal{L}_{\text{hcons}} = \frac{1}{N}\sum_{i=1}^{N} \text{KL}\Big(\text{softmax}\left(\text{sg}[u(x_i)]/T\right) \parallel \text{softmax}\left(\text{Agg}(s(x_i))/T\right)\Big), \tag{11}$$

where $T > 0$ is a temperature parameter and $\text{sg}[\cdot]$ denotes stop-gradient. The stop-gradient operation makes the supervised basic branch a stable abstraction target, preventing the consistency loss from destabilizing the family-level classifier.

Let $\mathcal{L}_{\text{fine}}$ denote the original fine-grained training objective. The Stage 1 objective is

$$\mathcal{L}_{\text{stage1}} = \mathcal{L}_{\text{fine}} + \lambda_{\text{b}}(t)\mathcal{L}_{\text{basic}} + \lambda_{\text{h}}(t)\mathcal{L}_{\text{hcons}}, \tag{12}$$

where $\lambda_{\text{b}}(t)$ and $\lambda_{\text{h}}(t)$ are scheduled over training.

The basic-level branch is not used to replace the fine-grained predictor. Instead, it provides a supervised abstraction target that regularizes how the fine-grained predictor organizes classes at the family level. In practice, we use an abstraction-first schedule: direct basic-level supervision is emphasized early, allowing the model to acquire explicit family-level structure, while hierarchical consistency is strengthened later, after the fine-grained predictor becomes more stable. Full schedules and implementation details are provided in Appendix D.

At the end of Stage 1, the model provides two complementary outputs: a fine-grained predictor for the original recognition task and a basic-level branch that supplies explicit family-level confidence. Stage 2 uses these outputs to determine when additional within-family residual refinement should be trained and activated.

3.3 Stage 2: Commitment-Guided Selective Differentiation

Stage 2 introduces sibling-level residual refinement. The key idea is that additional fine-grained differentiation should be trained and activated only when the model has established sufficiently reliable family-level structure. To stabilize this process, the best Stage 1 checkpoint is frozen as a teacher, and the student is initialized from the same checkpoint. The teacher remains fixed during Stage 2 and provides stable commitment signals for selecting samples for training residual refinement.

The Stage 2 student produces fine-grained, basic-level, and residual logits:

$$s^{\text{stu}}(x) \in \mathbb{R}^K, \quad u^{\text{stu}}(x) \in \mathbb{R}^M, \quad \text{and} \quad r(x) \in \mathbb{R}^K. \tag{13}$$

The residual logits are interpreted as a family-conditioned correction rather than as a standalone classifier. For a family b, let

$$\mathcal{I}_b = \{y \in \mathcal{Y} \mid g(y) = b\} \tag{14}$$

denote its sibling set. We define the centered family residual as

$$\tilde{r}_y(x; b) = \begin{cases} r_y(x) - \frac{1}{|\mathcal{I}_b|}\sum_{j\in\mathcal{I}_b} r_j(x), & y \in \mathcal{I}_b, \\ 0, & y \notin \mathcal{I}_b. \end{cases} \tag{15}$$

This centering and masking design constrains the residual branch to model relative sibling-level differences. It cannot directly increase or decrease scores outside the committed family, and therefore acts as a local correction on top of the base predictor rather than as a second global classifier.

Teacher-based commitment for training. For a training sample (x_i, y_i) with $b_i = g(y_i)$, the frozen teacher produces fine-grained logits $s^{\mathrm{tea}}(x_i)$ and basic-level logits $u^{\mathrm{tea}}(x_i)$. We define the target-aware teacher commitment event as

$$C_\tau^{\mathrm{train}}(x_i) = \mathbb{I}\Big[\,\mathrm{softmax}\left(u^{\mathrm{tea}}(x_i)\right)_{b_i} \geq \tau \;\wedge\; \arg\max_{b \in \mathcal{B}} \mathrm{Agg}\left(s^{\mathrm{tea}}(x_i)\right)_b = b_i\Big], \tag{16}$$

where τ is the training commitment threshold.

The two conditions in Eq. (16) serve different purposes. The basic-level confidence condition requires the teacher to assign high probability to the ground-truth family, while the agreement condition requires the family-level prediction induced by the teacher's fine-grained logits to be consistent with that family. Using the frozen teacher, rather than the evolving student, prevents the residual branch from changing the sample selection rule during Stage 2 training.

Refinement is trained primarily for medium-shot and few-shot classes. The Stage 2 training mask is

$$m_i = C_\tau^{\mathrm{train}}(x_i) \cdot \mathbb{I}[y_i \in \mathcal{Y}_{\mathrm{mf}}]. \tag{17}$$

This mask selects samples whose family-level structure is reliable according to the frozen teacher and whose fine-grained labels belong to the target lower-frequency regimes.

For selected samples, let $\pi_b : \mathcal{I}_b \to \{1, \ldots, |\mathcal{I}_b|\}$ map each fine label to its local sibling index. The residual objective is

$$\mathcal{L}_{\mathrm{sibling}} = \frac{\sum_i m_i \; \mathrm{CE}\Big(\tilde{r}_{\mathcal{I}_{b_i}}(x_i; b_i), \pi_{b_i}(y_i)\Big)}{\sum_i m_i + \epsilon}. \tag{18}$$

Here, $\epsilon > 0$ provides numerical stability, and unselected samples contribute zero. Subtracting the family mean leaves within-family softmax probabilities unchanged; centering removes a shared offset when merging residuals, while masking leaves outside-family logits unchanged. The parameter-space residual penalties are specified in Appendix D.3.

Stabilized Stage 2 objective. In addition to the selective sibling-level loss, Stage 2 uses teacher-student distillation and residual regularization to preserve the Stage 1 structure and keep the correction conservative. We write the stabilization term compactly as

$$\mathcal{L}_{\mathrm{stab}} = \lambda_{\mathrm{df}} \mathcal{L}_{\mathrm{distill}}^{\mathrm{fine}} + \lambda_{\mathrm{db}} \mathcal{L}_{\mathrm{distill}}^{\mathrm{basic}} + \lambda_{\mathrm{r}} \mathcal{L}_{\mathrm{reg}}, \tag{19}$$

where the distillation and regularization terms are defined in Appendix D. The Stage 2 objective is

$$\mathcal{L}_{\mathrm{stage2}} = \mathcal{L}_{\mathrm{fine}} + \eta_{\mathrm{b}}(t)\mathcal{L}_{\mathrm{basic}} + \eta_{\mathrm{h}}(t)\mathcal{L}_{\mathrm{hcons}} + \lambda_{\mathrm{s}}(t)\mathcal{L}_{\mathrm{sibling}} + \mathcal{L}_{\mathrm{stab}}. \tag{20}$$

In all experiments, $\mathcal{L}_{\mathrm{fine}}$, $\mathcal{L}_{\mathrm{basic}}$, and $\mathcal{L}_{\mathrm{hcons}}$ are computed on the student base prediction pathway before residual merging. The selective sibling loss $\mathcal{L}_{\mathrm{sibling}}$ directly supervises the centered residual branch on committed samples. This separation keeps the residual branch as a conditional within-family correction rather than an unconstrained global classifier. The retained abstraction losses help the student preserve family-level structure, while the stabilization term reduces semantic drift during residual refinement. Detailed schedules, regularizers, and training algorithms are provided in Appendix D.

3.4 Training, Inference, and Interpretation

The complete training pipeline consists of two stages. Stage 1 trains the fine-grained predictor and the basic-level branch using Eq. (12). Stage 2 initializes a student from the best Stage 1 checkpoint, freezes the same checkpoint as a teacher, and optimizes the student using Eq. (20). The teacher is used only during Stage 2 training to provide stable commitment signals and distillation targets. At test time, only the trained student is used.

Inference with selective residual correction. At inference time, labels are unavailable, so the target-aware training commitment in Eq. (16) cannot be used. Instead, the trained student evaluates a prediction-based commitment event. Let $s_{\mathrm{base}}(x)$ denote the student fine-grained logits before residual merging. The student first predicts the family from its basic-level branch and from the aggregated fine-grained logits:

$$\hat{b}_u(x) = \arg\max_{b \in \mathcal{B}} u_b^{\mathrm{stu}}(x), \qquad \hat{b}_s(x) = \arg\max_{b \in \mathcal{B}} \mathrm{Agg}\left(s_{\mathrm{base}}(x)\right)_b. \tag{21}$$

The evaluation-time commitment event is

$$C^{\mathrm{eval}}_{\tau_{\mathrm{eval}}}(x) = \mathbb{I}\Big[\max_{b \in \mathcal{B}} \mathrm{softmax}\left(u^{\mathrm{stu}}(x)\right)_b \geq \tau_{\mathrm{eval}} \ \wedge \ \hat{b}_u(x) = \hat{b}_s(x)\Big], \tag{22}$$

where τ_{eval} is the evaluation commitment threshold.

Since Stage 2 refinement is trained for medium-shot and few-shot classes, the evaluation mask also checks whether the base prediction belongs to the target lower-frequency set:

$$\hat{y}_{\mathrm{base}}(x) = \arg\max_{y \in \mathcal{Y}} s_{\mathrm{base},y}(x), \tag{23}$$

$$m^{\mathrm{eval}}(x) = C^{\mathrm{eval}}_{\tau_{\mathrm{eval}}}(x) \cdot \mathbb{I}[\hat{y}_{\mathrm{base}}(x) \in \mathcal{Y}_{\mathrm{mf}}]. \tag{24}$$

The final logits are obtained by applying the centered residual correction within the committed family:

$$s^{\mathrm{final}}(x) = s_{\mathrm{base}}(x) + \alpha_{\mathrm{r}}\, m^{\mathrm{eval}}(x)\, \tilde{r}(x; \hat{b}_u(x)). \tag{25}$$

If either the commitment event fails or the base prediction is not in the target low-frequency set, then $m^{\mathrm{eval}}(x) = 0$ and the model falls back to $s_{\mathrm{base}}(x)$. Therefore, residual refinement is conservative at inference time: it is used only when the student predicts a reliable family and the prediction lies in the regime for which Stage 2 refinement was trained. Although training uses target-aware teacher commitment and inference relies on prediction-based student commitment, both rules preserve the same reliability principles: high family-level confidence and coarse agreement. Appendix F.7 and Appendix G.6 analyze this selection mismatch and the resulting coverage–reliability trade-off.

Interpretation. The design can be interpreted through the decomposition

$$p(y \mid x) = p(g(y) \mid x)\, p(y \mid g(y), x). \tag{26}$$

This identity separates fine-grained recognition into a family-level abstraction term and a within-family differentiation term. Under long-tailed scarcity, these two terms can have different reliability: family-level abstraction may be supported by examples from related classes, while sibling-level differentiation may remain fragile for rare classes.

BASICLT operationalizes this view by separating the learning process into two stages. Stage 1 targets the family-level term through basic-level supervision and hierarchical consistency. Stage 2 refines the within-family term only when family commitment is reliable. Thus, the method does not simply add another classifier or increase discrimination everywhere; instead, it regulates the semantic granularity at which additional discrimination is introduced. Appendix B provides a fuller formal discussion.

4 Experiments

We evaluate BASICLT from three perspectives: overall performance on standard long-tailed benchmarks, the contribution of its main components, and whether the empirical behavior is consistent with the proposed abstraction-to-differentiation mechanism.

4.1 Experimental Setup

Datasets and metrics. We evaluate on CIFAR-100-LT, CIFAR-10-LT [1, 2], ImageNet-LT [20], iNaturalist 2018 [21], and Places-LT [22]. We report top-1 accuracy and, when supported by the benchmark protocol, many-shot, medium-shot, and few-shot accuracy.

Table 1: CIFAR-100-LT results under three imbalance ratios. Best results are in **bold**, and second-best results are underlined. *: reported results; †: reproduced results.

Method	Source	IR = 100				IR = 50				IR = 10			
		Many	Medium	Few	Overall	Many	Medium	Few	Overall	Many	Medium	Few	Overall
CE-DRW [2]*	*NeurIPS 19*	63.4	41.2	15.7	41.4	60.6	39.0	22.9	45.0	62.5	48.6	-	58.2
LDAM-DRW [2]*	*NeurIPS 19*	61.2	39.5	14.4	42.0	64.9	45.1	28.7	46.6	70.3	57.0	-	58.7
BS [23]*	*NeurIPS 20*	60.8	44.6	20.4	46.1	66.0	47.2	30.1	50.9	68.0	56.0	-	61.0
cRT [3]*	*ICLR 20*	61.8	46.2	27.4	47.3	65.4	52.1	32.5	51.7	73.3	67.5	-	63.2
LADE [24]*	*CVPR 21*	60.2	43.9	19.6	45.6	65.5	46.3	29.7	50.1	69.0	57.8	-	61.6
MiSLAS [25]*	*CVPR 21*	62.1	45.3	21.1	46.8	66.8	48.4	31.2	51.5	70.0	58.5	-	62.5
RIDE (3 experts) [5]*	*ICLR 21*	64.7	46.5	22.3	48.0	67.5	50.8	27.9	51.7	66.4	49.4	-	61.8
BCL [6]*	*CVPR 22*	63.1	42.9	23.9	44.2	61.6	43.1	34.3	49.1	62.2	51.8	-	58.9
SADE [26]*	*NeurIPS 22*	63.4	49.2	25.1	49.8	67.2	53.5	31.8	53.8	72.9	62.5	-	63.6
BalPoE [27]*	*CVPR 23*	65.3	51.1	28.0	52.0	69.2	55.9	34.5	56.3	73.0	63.2	-	64.8
ConCutmix [28]†	*TIP 24*	67.0	52.0	29.2	53.2	69.0	55.6	38.0	57.4	73.0	63.0	-	64.5
LSC [29]*	*ICML 24*	66.0	50.0	27.0	51.8	69.0	55.0	36.0	56.5	73.6	63.8	-	65.0
PRL [30]*	*NeurIPS 24*	68.7	55.3	31.2	52.8	67.5	53.5	39.0	56.6	74.1	64.5	-	65.6
ProCo [31]†	*TPAMI 24*	69.0	52.7	34.1	52.8	69.1	55.2	37.0	57.1	73.9	64.0	-	65.5
ICL [32]†	*IJCAI 25*	70.6	61.1	36.5	57.6	69.1	56.8	38.9	61.3	73.9	66.8	-	69.3
LOS [33]†	*ICLR 25*	68.6	53.8	30.2	54.9	69.1	55.8	38.4	58.8	74.0	64.3	-	69.7
FeatRecon [34]†	*ICLR 25*	68.2	53.3	34.0	52.5	68.8	54.3	37.2	57.0	73.4	63.5	-	65.3
IP-TSL [35]†	*NeurIPS 25*	62.4	59.7	31.9	52.4	68.9	55.0	37.5	55.9	73.5	64.1	-	62.6
BASICLT (Ours)	-	**78.4**±0.3	**63.2**±0.4	**37.0**±0.5	**60.6**±0.4	**77.4**±0.4	**59.6**±0.4	**53.1**±0.5	**65.7**±0.3	**76.2**±0.5	**68.3**±0.3	-	**73.8**±0.5

Basic taxonomy. Each dataset uses a fixed fine-to-family mapping. CIFAR-100-LT uses the official superclasses; CIFAR-10-LT uses a WordNet-derived two-family grouping; ImageNet-LT uses a fixed curated WordNet family taxonomy; iNaturalist 2018 uses the public biological hierarchy; and Places-LT uses an audited scene-semantic grouping. The mapping is used only for abstraction and selective refinement, while evaluation remains on the original fine-grained label space. All taxonomies are fixed before training and are not tuned using validation or test accuracy. Details and sensitivity analyses are provided in Appendix C and Appendix F.16. Additional taxonomy-matched controls are provided in Appendix H to separate the effect of access to the taxonomy from the proposed abstraction-to-differentiation design.

Training and comparison. BASICLT is trained using the two-stage procedure described in Section 3. Stage 1 learns basic-level abstraction, and Stage 2 trains teacher-guided residual refinement. Full schedules, thresholds, and implementation details are provided in Appendix D. We compare with representative long-tailed recognition methods. Results marked with * are from original papers, and † denotes reproduced results. Unless otherwise specified, all BASICLT results are reported as mean ± standard deviation over five independent runs.

4.2 Main Results

Tables 1 and 2 summarize broad benchmark comparisons, where BASICLT achieves competitive overall performance and consistent medium/few-shot gains. Since these tables mix reported and reproduced baselines, controlled attribution is provided in Section 4.3 and Appendix H.

On CIFAR-100-LT, BASICLT improves overall accuracy across all imbalance ratios, with clear medium/few-shot gains. Refinement eligibility uses the *base-predicted* shot group, whereas reported group accuracy uses the *ground-truth* class. A true many-shot example initially predicted as medium/few can therefore be corrected even though many-shot base predictions are not refined. Many-shot gains reflect both the improved base pathway and corrections of eligible errors; Appendix H.6 separates these effects.

Across CIFAR-10-LT, ImageNet-LT, iNaturalist 2018, and Places-LT, BASICLT remains competitive with strong prior methods and often improves the medium-shot and few-shot regimes. ImageNet-LT results are particularly informative: although the many-shot accuracy is slightly below the strongest prior result, BASICLT achieves the best medium-shot, few-shot, and overall accuracy, suggesting its benefit in lower-frequency regimes. The improved iNaturalist 2018 and Places-LT results further indicate that the framework can be applied to fine-grained biological recognition and scene-centric long-tailed recognition.

4.3 Ablation and Analysis

We conduct ablations on CIFAR-100-LT (IR=100) using the same data and taxonomy unless otherwise specified. Full results, including training-budget and parameter-matched controls, residual variants,

Table 2: Results on CIFAR-10-LT, ImageNet-LT, iNaturalist 2018, and Places-LT. Best results are in **bold**, and second-best results are underlined. *: reported results; †: reproduced results.

Method	Source	CIFAR-10-LT (IR=100)				ImageNet-LT				iNat. 2018	Places-LT
		Many	Medium	Few	All	Many	Medium	Few	All	All	All
Cross-Entropy	-	90.5	62.1	37.8	70.4	40.9	10.7	0.4	41.6	61.7	27.4
LDAM-DRW [2]*	*NeurIPS 19*	83.3	73.5	72.5	77.0	60.4	46.9	30.7	48.8	64.6	30.5
BS [23]*	*NeurIPS 20*	88.5	76.2	68.3	79.1	62.2	48.8	38.6	51.4	70.6	39.4
cRT [3]*	*ICLR 20*	88.7	76.5	63.8	78.4	61.8	46.2	27.3	50.3	65.2	33.2
MiSLAS [25]*	*CVPR 21*	89.7	79.2	73.1	82.1	61.7	51.3	35.8	52.7	70.7	33.5
RIDE (3 experts) [5]*	*ICLR 21*	96.1	89.0	76.5	87.2	66.2	52.3	36.5	55.4	68.8	40.3
BCL [6]*	*CVPR 22*	92.9	83.3	55.3	81.2	66.5	53.6	30.7	51.8	71.8	-
SADE [26]*	*NeurIPS 22*	95.8	88.5	76.4	86.9	67.2	54.7	35.1	56.1	72.7	40.9
BalPoE [27]*	*CVPR 23*	96.3	89.2	76.4	87.3	68.3	57.2	40.5	58.5	75.0	40.8
LSC [29]*	*ICML 24*	92.5	82.0	73.5	83.5	66.0	50.0	27.0	56.5	73.9	-
PRL [30]*	*NeurIPS 24*	93.8	84.1	74.5	84.5	**72.5**	58.2	41.8	60.8	75.1	41.6
ProCo [31]†	*TPAMI 24*	96.4	89.1	76.6	87.5	69.1	59.8	42.1	60.2	73.5	-
ICL [32]†	*IJCAI 25*	96.9	88.5	76.3	87.1	68.2	59.1	45.8	60.2	75.9	42.1
LOS [33]*	*ICLR 25*	92.0	83.5	74.2	84.8	-	-	-	54.4	70.8	41.9
FeatRecon [34]*	*ICLR 25*	94.5	86.8	74.3	85.2	-	-	-	56.8	72.9	42.3
IP-TSL [35]†	*NeurIPS 25*	82.1	77.2	66.5	76.4	61.2	50.8	36.5	51.7	74.0	-
BASICLT (Ours)	-	**97.1**±0.3	**89.9**±0.2	**77.2**±0.6	**87.8**±0.4	70.4±0.4	**62.4**±0.2	**49.5**±0.5	**63.2**±0.5	**77.2**±0.5	**44.5**±0.4

Table 3: Core ablation on **CIFAR-100-LT** with IR= 100. Results are reported as mean ± standard deviation over five runs.

Variant	Many	Med.	Few	All
Base predictor	69.5±0.4	52.4±0.5	30.2±0.6	52.6±0.2
+ Basic branch	70.3±0.5	54.1±0.3	31.8±0.7	54.2±0.5
+ Stage 1 full	72.1±0.3	56.4±0.6	33.5±0.6	56.1±0.4
+ Stage 2 w/o distill.	66.8±0.4	58.2±0.2	35.1±0.3	55.4±0.1
+ Stage 2 w/ distill.	75.3±0.2	60.8±0.4	35.8±0.5	58.7±0.5
Full BASICLT	**78.4**±0.3	**63.2**±0.4	**37.0**±0.5	**60.6**±0.4

Table 4: Refinement policy ablation on **CIFAR-100-LT** with IR= 100. Results are reported as mean ± standard deviation over five runs.

Policy	Many	Med.	Few	All
Stage 1 only	72.1±0.4	56.4±0.5	33.5±0.2	56.1±0.5
Uniform refinement	70.4±0.5	57.0±0.4	34.2±0.3	55.8±0.2
Confidence-only	74.8±0.3	59.3±0.1	35.2±0.4	58.2±0.4
Agreement-only	75.5±0.5	60.2±0.3	35.9±0.2	59.0±0.3
Commitment-guided	**78.4**±0.3	**63.2**±0.4	**37.0**±0.5	**60.6**±0.4

and sensitivity analyses, are provided in Appendix F. These controls suggest that the observed gain is not explained by longer training or simply adding more model parameters, but is mainly associated with the proposed abstraction-to-differentiation design.

Table 3 shows that both stages contribute: Stage 1 improves all shot groups, while Stage 2 stabilization yields a better trade-off than distillation-free refinement. The row "+ Stage 2 w/ distill." reports the stabilized base pathway before evaluation-time residual merging; "Full BASICLT" further applies commitment-guided residual correction to eligible medium/few-shot predictions. Appendix H.3 decomposes these factors.

Table 4 shows that selective activation is crucial: uniform refinement is weaker, and confidence-only or agreement-only variants lag behind the full commitment rule.

Figure 2 supports the abstraction-to-differentiation mechanism: family-level prediction is more reliable under scarcity, Stage 1 improves coarse consistency, and Stage 2 activates refinement only on a limited committed subset. The threshold curve further shows a coverage–reliability trade-off, indicating that BASICLT controls when fine-grained differentiation is used rather than simply adding capacity. The teacher is training-only; overhead and additional controls on taxonomy sensitivity, training budget, wrong-family commitment, and taxonomy-matched baselines are reported in Appendices F–H.

5 Conclusion

We presented BASICLT, a two-stage framework for long-tailed recognition that controls the semantic granularity of discrimination under scarce supervision. BASICLT first learns basic-level abstraction and then applies family-conditioned residual refinement only under reliable commitment. Experiments across five benchmarks show competitive performance, with consistent gains in medium-shot and few-shot regimes. Ablations and analyses further demonstrate that both abstraction learning and commitment-guided selective refinement are important, suggesting that granularity control is a useful complement to conventional long-tailed rebalancing strategies.

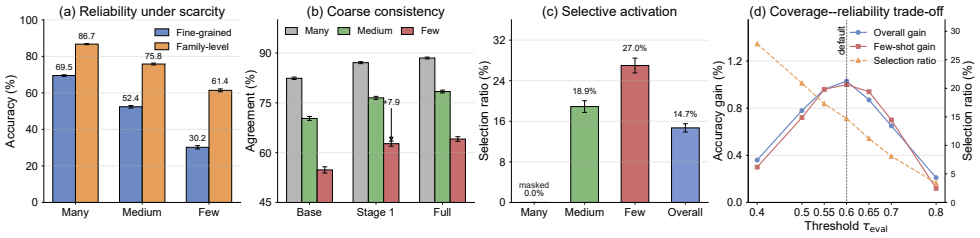


Figure 2: Mechanism analysis of BASICLT. (a) Family-level prediction is more reliable than fine-grained prediction under scarcity. (b) Stage 1 improves coarse semantic consistency. (c) Stage 2 activates residual refinement only on a limited committed subset, grouped by the base-predicted shot regime. (d) The evaluation threshold reveals a coverage–reliability trade-off between accuracy gain and selection ratio, with the default threshold as a balanced operating point.

Acknowledgments

The authors gratefully acknowledge the support from the National Natural Science Foundation of China (NSFC) under Grant Nos. 62402472, 12227901, and 62476109. This work was also supported by the Natural Science Foundation of Jiangsu Province (No. BK20240461), the Project of Stable Support for Youth Team in Basic Research Field, CAS (No. YSBR-005), and the Academic Leaders Cultivation Program at USTC. The AI-driven experiments, simulations, and model training were performed on the robotic AI-Scientist platform of the Chinese Academy of Sciences.

Competing Interests. The authors declare that they have no competing interests.

References

[1] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277, 2019.

[2] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32, 2019.

[3] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019.

[4] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. *arXiv preprint arXiv:2007.07314*, 2020.

[5] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella X Yu. Long-tailed recognition by routing diverse distribution-aware experts. *arXiv preprint arXiv:2010.01809*, 2020.

[6] Jianggang Zhu, Zheng Wang, Jingjing Chen, Yi-Ping Phoebe Chen, and Yu-Gang Jiang. Balanced contrastive learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6908–6917, 2022.

[7] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9719–9728, 2020.

[8] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 715–724, 2021.

[9] Yuan Dong, Zhe Zhao, Liheng Yu, Di Wu, and Pengkun Wang. Guide: Gated uncertainty-informed disentangled experts for long-tailed recognition. In *International Conference on Learning Representations*, volume 2026, pages 26004–26034, 2026.

[10] Yuan Dong, Zhe Zhao, Liheng Yu, Di Wu, Pengkun Wang, and Yang Wang. Manidream: Manifold completion via prototype-anchored dreaming for long-tailed visual recognition. *International Journal of Computer Vision*, 134(8):386, 2026.

[11] Zhuang Qi, Lei Meng, Zhaochuan Li, Han Hu, and Xiangxu Meng. Cross-silo feature space alignment for federated learning on clients with imbalanced data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 19986–19994, 2025.

[12] Zhuang Qi, Yu Pan, Lei Meng, Sijin Zhou, Han Yu, Xiaoxiao Li, and Xiangxu Meng. Global prompt refinement with non-interfering attention masking for one-shot federated learning. *Advances in Neural Information Processing Systems*, 38:12121–12145, 2026.

[13] Zhuang Qi, Ying-Peng Tang, Lei Meng, Guoqing Chao, Lei Wu, Han Yu, and Xiangxu Meng. From selection to scheduling: Federated geometry-aware correction makes exemplar replay work better under continual dynamic heterogeneity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17494–17504, June 2026.

[14] Zhicheng Yan, Hao Zhang, Robinson Piramuthu, Vignesh Jagadeesh, Dennis DeCoste, Wei Di, and Yizhou Yu. Hd-cnn: hierarchical deep convolutional neural networks for large scale visual recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 2740–2748, 2015.

[15] Yanming Guo, Yu Liu, Erwin M Bakker, Yuanhao Guo, and Michael S Lew. Cnn-rnn: a large-scale hierarchical image classification framework. *Multimedia tools and applications*, 77 (8):10251–10271, 2018.

[16] Luca Bertinetto, Romain Mueller, Konstantinos Tertikas, Sina Samangooei, and Nicholas A Lord. Making better mistakes: Leveraging class hierarchies with deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12506–12515, 2020.

[17] Yingjun Du, Jiayi Shen, Xiantong Zhen, and Cees GM Snoek. Superdisco: Super-class discovery improves visual recognition for the long-tail. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19944–19954. IEEE, 2023.

[18] Eleanor Rosch, Carolyn B Mervis, Wayne Gray, David Johnson, and Penny Boyes-Braem. Basic objects in natural categories. *Cognitive psychology: Key readings*, 448, 2004.

[19] Michelle R Greene and Alyssa Magill Rohan. The brain prioritizes the basic level of object category abstraction. *Scientific Reports*, 15(1):31, 2025.

[20] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2537–2546, 2019.

[21] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778, 2018.

[22] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.

[23] Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. *Advances in neural information processing systems*, 33:4175–4186, 2020.

[24] Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling label distribution for long-tailed visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6626–6636, 2021.

[25] Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia. Improving calibration for long-tailed recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16489–16498, 2021.

[26] Yifan Zhang, Bryan Hooi, Lanqing Hong, and Jiashi Feng. Self-supervised aggregation of diverse experts for test-agnostic long-tailed recognition. *Advances in neural information processing systems*, 35:34077–34090, 2022.

[27] Emanuel Sanchez Aimar, Arvi Jonnarth, Michael Felsberg, and Marco Kuhlmann. Balanced product of calibrated experts for long-tailed recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19967–19977, 2023.

[28] Haolin Pan, Yong Guo, Mianjie Yu, and Jian Chen. Enhanced long-tailed recognition with contrastive cutmix augmentation. *IEEE Transactions on Image Processing*, 33:4215–4230, 2024.

[29] Tong Wei, Zhen Mao, Zi-Hao Zhou, Yuanyu Wan, and Min-Ling Zhang. Learning label shift correction for test-agnostic long-tailed recognition. In *Forty-first International Conference on Machine Learning*, 2024.

[30] Zhe Zhao, HaiBin Wen, Zikang Wang, Pengkun Wang, Fanfu Wang, Song Lai, Qingfu Zhang, and Yang Wang. Breaking long-tailed learning bottlenecks: A controllable paradigm with hypernetwork-generated diverse experts. *Advances in Neural Information Processing Systems*, 37:7493–7520, 2024.

[31] Chaoqun Du, Yulin Wang, Shiji Song, and Gao Huang. Probabilistic contrastive learning for long-tailed visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(9):5890–5904, 2024.

[32] Yanhao Chen, Zhongquan Jian, Nianxin Ke, Shuhao Hu, Junjie Jiao, Qingqi Hong, and Qingqiang Wu. Enhancing mixture of experts with independent and collaborative learning for long-tail visual recognition. In James Kwok, editor, *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 828–836. International Joint Conferences on Artificial Intelligence Organization, 8 2025. doi: 10.24963/ijcai.2025/93. Main Track.

[33] Siyu Sun, Han Lu, Jiangtong Li, Yichen Xie, Tianjiao Li, Xiaokang Yang, Liqing Zhang, and Junchi Yan. Rethinking classifier re-training in long-tailed recognition: Label over-smooth can balance. In *The Thirteenth International Conference on Learning Representations*, 2025.

[34] Lingjie Yi, Jiachen Yao, Weimin Lyu, Haibin Ling, Raphael Douady, and Chao Chen. Geometry of long-tailed representation learning: Rebalancing features for skewed distributions. In *The Thirteenth International Conference on Learning Representations*, 2025.

[35] Fudong Lin and Xu Yuan. Long-tailed recognition via information-preservable two-stage learning. *arXiv preprint arXiv:2510.08836*, 2025.

[36] Zhe Zhao, Zhiheng Gong, Pengkun Wang, HaiBin Wen, Cankun Guo, Bo Xue, Xi Lin, Zhenkun Wang, Qingfu Zhang, and Yang Wang. Ts-mof: Two-stage multi-objective fine-tuning for long-tailed recognition. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

[37] Jun Li, Zichang Tan, Jun Wan, Zhen Lei, and Guodong Guo. Nested collaborative learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6949–6958, 2022.

[38] Tianhong Li, Peng Cao, Yuan Yuan, Lijie Fan, Yuzhe Yang, Rogerio S Feris, Piotr Indyk, and Dina Katabi. Targeted supervised contrastive learning for long-tailed recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6918–6928, 2022.

[39] Jiaan Luo, Feng Hong, Qiang Hu, Xiaofeng Cao, Feng Liu, and Jiangchao Yao. Long-tailed recognition with model rebalancing. *arXiv preprint arXiv:2510.08177*, 2025.

[40] Seulki Park, Youngkyu Hong, Byeongho Heo, Sangdoo Yun, and Jin Young Choi. The majority can help the minority: Context-rich minority oversampling for long-tailed classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6887–6896, 2022.

[41] Jie Shao, Ke Zhu, Hanxiao Zhang, and Jianxin Wu. Diffult: Diffusion for long-tail recognition without external knowledge. *Advances in Neural Information Processing Systems*, 37:123007–123031, 2024.

[42] Carlos N Silla Jr and Alex A Freitas. A survey of hierarchical classification across different application domains. *Data mining and knowledge discovery*, 22(1):31–72, 2011.

[43] Wenhao Wang, Yifan Sun, Wei Li, and Yi Yang. Transhp: Image classification with hierarchical prompting. *Advances in Neural Information Processing Systems*, 36:28187–28200, 2023.

[44] Hyeongjun Kwon, Jinhyun Jang, Jin Kim, Kwonyoung Kim, and Kwanghoon Sohn. Improving visual recognition with hyperbolical visual hierarchy mapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17364–17374, 2024.

[45] Seulki Park, Youren Zhang, Stella X Yu, Sara Beery, and Jonathan Huang. Visually consistent hierarchical image classification. *arXiv preprint arXiv:2406.11608*, 2024.

[46] J David Smith. Prototypes, exemplars, and the natural history of categorization. *Psychonomic bulletin & review*, 21(2):312–331, 2014.

[47] Caitlin R Bowman, Takako Iwashita, and Dagmar Zeithamova. Tracking prototype and exemplar representations in the brain across learning. *elife*, 9:e59360, 2020.

[48] Vanessa E Ghosh and Asaf Gilboa. What is a memory schema? a historical perspective on current neuroscience literature. *Neuropsychologia*, 53:104–114, 2014.

[49] Asaf Gilboa and Hannah Marlatte. Neurobiology of schemas and schema-mediated memory. *Trends in cognitive sciences*, 21(8):618–631, 2017.

[50] Troy M Houser. A boundedly rational model for category learning. *Frontiers in Psychology*, 15: 1477514, 2024.

[51] Oded Bein and Yael Niv. Schemas, reinforcement learning and the medial prefrontal cortex. *Nature Reviews Neuroscience*, 26(3):141–157, 2025.

Appendix

BASICLT: Basic-Level Abstraction and Selective Differentiation for Long-Tailed Recognition

The appendix is organized as follows. Sections A–D provide related work, formal details, taxonomy construction, and implementation. Sections E–H report evaluation scope, ablations, mechanism analyses, and attribution controls. Section I discusses limitations and broader impacts.

Section	Content
A	Extended related work
B	Extended formulation and theory
C	Dataset-specific taxonomy construction and assumptions
D	Implementation, dataset-wise configurations, and regularization
E	Extended results and evaluation scope
F	Extended ablation studies
G	Extended analysis and visualization
H	Taxonomy-matched controls and Stage 2 attribution
I	Limitations and broader impacts

A Extended Related Work

This appendix expands the related work discussion in the main text. We focus on three lines of work that are most relevant to BASICLT: long-tailed recognition, hierarchy-aware visual recognition, and cognitively motivated category learning. These areas address different aspects of recognition under limited or structured supervision. Long-tailed recognition studies how to learn from imbalanced class distributions; hierarchy-aware recognition studies how semantic structure can be incorporated into visual prediction; and basic-level category learning suggests that intermediate abstraction can provide a useful basis for subsequent fine discrimination. BASICLT lies at the intersection of these perspectives, but differs from prior work by using basic-level structure to control when fine-grained differentiation should be introduced under scarcity.

A.1 Additional Long-Tailed Recognition Methods

Long-tailed recognition has been widely studied as a problem of learning under highly imbalanced class distributions. A common line of work modifies the optimization objective to reduce the dominance of head classes. Representative examples include class-balanced loss based on the effective number of samples [1], margin-based reweighting and deferred re-balancing in LDAM-DRW [2], balanced posterior correction in Balanced Softmax [23], and prior-aware logit adjustment [4]. These methods improve tail recognition by correcting the imbalance in gradients, logits, or posterior estimates. However, they generally retain a flat fine-grained label space in which all classes compete at the same semantic resolution.

Another important direction is decoupled training. Decoupling representation learning and classifier learning has been shown to be effective for long-tailed recognition [3]. This perspective has motivated a broad family of two-stage or post-hoc correction methods. For example, classifier retraining can improve tail performance by reducing classifier bias, while more recent two-stage methods further study how to preserve discriminative information or rebalance the classifier after representation learning [33, 35, 36]. These approaches demonstrate the value of separating different learning objectives across stages. In contrast, BASICLT separates not only optimization stages, but also semantic granularity: Stage 1 emphasizes basic-level abstraction, while Stage 2 introduces sibling-level differentiation only under reliable family-level commitment.

Architectural specialization provides another influential approach to long-tailed recognition. BBN combines conventional and re-balanced branches to improve head–tail trade-offs [7]. RIDE improves long-tailed recognition through diverse distribution-aware experts [5]. Subsequent expert-based methods further explore collaboration, self-supervision, uncertainty, and specialization under skewed class distributions [37, 26, 30]. Other methods improve long-tailed representation geometry through contrastive learning, hyperspherical constraints, feature reconstruction, or model-space rebalancing [6,

38, 34, 39]. Data-centric approaches, such as mixup-based strategies and generative augmentation, seek to enrich tail supervision by synthesizing or combining training examples [40, 41].

These developments have substantially advanced long-tailed recognition. Nevertheless, most of them address imbalance while preserving a uniform leaf-level discrimination objective. Their interventions differ in sampling, loss design, classifier correction, representation geometry, or model architecture, but the final prediction space is typically treated as flat. BASICLT studies a complementary question: while retaining fine-grained supervision, when should extra sibling-specialized capacity be introduced under scarcity? Stage 1 jointly learns the target fine-grained predictor and its family scaffold. Stage 2 subsequently trains and activates additional residual differentiation under reliable commitment. This temporal ordering concerns the added refinement capacity, not the first use of fine labels.

A.2 Additional Hierarchy-Aware Recognition Methods

Hierarchy-aware classification has a long history in machine learning and computer vision. In hierarchical classification, labels are organized by semantic relations, and prediction errors can have different meanings depending on their distance in the hierarchy [42]. In visual recognition, early hierarchical deep models such as HD-CNN use coarse predictions to organize fine-level experts [14], while CNN-RNN models capture dependencies among category levels through recurrent prediction structures [15]. These methods show that semantic hierarchy can provide useful inductive bias when the label space is large or structured.

More recent work has studied how label hierarchies can shape visual prediction behavior. Making Better Mistakes encourages errors that are semantically closer in a hierarchy [16]. Other approaches incorporate ancestor information, hierarchical prompts, or structured embeddings to improve prediction consistency across semantic levels [43, 44]. Recent hierarchy-aware methods also emphasize that multi-level predictions should remain visually consistent, rather than relying only on label structure [45]. These studies demonstrate that hierarchical information can improve recognition, calibration, interpretability, and semantic consistency.

BASICLT is related to hierarchy-aware recognition because it uses a fixed mapping from fine-grained labels to basic-level families. However, the role of hierarchy in our framework is different. Most hierarchy-aware methods use taxonomy to define multi-level outputs, encourage semantically meaningful errors, or improve structured prediction. In BASICLT, the taxonomy is used as a control mechanism for long-tailed learning. The basic-level branch is not introduced merely to perform auxiliary coarse classification; it provides an intermediate abstraction layer whose reliability determines whether sibling-level refinement should be trained and activated. Thus, hierarchy is used to regulate the semantic granularity of discrimination under scarcity.

This distinction is important for interpreting the method. BASICLT does not replace fine-grained recognition with hierarchical prediction, nor does it require the final classifier to output a full taxonomy path. All final evaluations remain on the original fine-grained task. The basic-level structure is used during training and inference to determine when within-family differentiation is reliable enough to be useful. In this sense, BASICLT can be viewed as a granularity-aware refinement framework rather than a standard hierarchical classifier.

Superclass discovery and reliability-controlled use. SuperDisco [17] learns multi-level superclass graph representations and uses message passing to refine image features for long-tailed recognition. It establishes a direct precedent for exploiting intermediate abstraction in this setting. BASICLT assumes a fixed family mapping and studies how its learned reliability controls additional sibling differentiation. The teacher selects training examples, and the student activates a centered, family-masked correction with fallback at evaluation. A discovered hierarchy could provide a candidate scaffold, but combining superclass discovery with this pipeline is not evaluated here. The taxonomy-matched controls in Appendix H.2 compare specific mechanisms under the same mapping; they are not a numerical reproduction of SuperDisco.

A.3 Basic-Level Abstraction and Category Learning

The idea of a basic level of abstraction originates from cognitive studies of categorization. Classic work by Rosch and colleagues argued that human categories often have a preferred intermediate level that balances informativeness and cognitive efficiency [18]. For example, an object may be

recognized more naturally at a basic level before being distinguished at a more specific subordinate level. This observation suggests that recognition does not always begin at the finest available semantic resolution.

Later studies further examined how abstraction and specificity interact during learning. Prototype and exemplar theories suggest that category learning can involve both abstract summaries and instance-level distinctions [46, 47]. Research on schema learning similarly emphasizes that abstract structure can organize perception, memory, and inference across related experiences [48, 49]. Recent work also connects abstraction to resource constraints, hierarchical decision making, and efficient representation learning [50, 51]. Evidence from visual cognition suggests that intermediate-level category structure can play an important role in recognition [19].

Our work is inspired by this basic-level perspective, but it does not aim to model human categorization directly. Instead, we use basic-level abstraction as a computational principle for long-tailed recognition. In long-tailed data, rare classes may not provide enough examples for reliable fine-grained separation from visually similar sibling classes. However, they can still benefit from structure shared with related classes at a coarser semantic level. This motivates the abstraction-to-differentiation design of BASICLT: first establish a stable basic-level representation, and then introduce additional sibling-level refinement only when family-level commitment is reliable.

This view also clarifies why the basic-level branch in BASICLT is not simply an auxiliary classifier. If the branch were used only as an additional supervision signal, the method would reduce to multi-task learning with coarse labels. Instead, BASICLT uses the basic-level branch to guide the timing and scope of fine-grained differentiation. The learned family-level confidence determines which samples are eligible for residual refinement, and the residual branch is constrained to operate within the committed family. Thus, basic-level abstraction serves as both a representational scaffold and a control signal for selective differentiation.

Overall, prior work in long-tailed recognition, hierarchy-aware classification, and category learning provides complementary motivation for BASICLT. Long-tailed recognition highlights the difficulty of learning under class imbalance; hierarchy-aware recognition shows that semantic structure can improve visual prediction; and basic-level category learning suggests that intermediate abstraction can support later fine discrimination. BASICLT combines these ideas in a unified framework for semantic granularity control under scarcity.

B Extended Formulation and Theory

This appendix provides additional formal details for BASICLT. The purpose is to clarify the notation, the commitment mechanism, and the probabilistic interpretation behind the abstraction-to-differentiation design. We emphasize that the formal discussion is intended to make the design logic explicit under simple assumptions, rather than to provide a complete theory of long-tailed recognition. The main method is described in Section 3; this appendix complements it by spelling out the basic taxonomy, the training- and evaluation-time commitment events, and the hierarchical decomposition used to motivate the staged framework.

B.1 Extended Notation and Problem Definition

The fine labels $\mathcal{Y}$, family labels $\mathcal{B}$, mapping g, and logits s, u, r are defined in Section 3. Each fine class has exactly one family, so the sibling sets form a partition. This assumption is used by aggregation, commitment, and family masking.

Definition B.1 (Basic taxonomy)**.** A basic taxonomy is a tuple $(\mathcal{Y}, \mathcal{B}, g)$ with a fixed mapping $g : \mathcal{Y} \to \mathcal{B}$. Its nonempty sibling sets $\mathcal{I}_b = \{y : g(y) = b\}$ are disjoint and cover $\mathcal{Y}$. The taxonomy is an intermediate scaffold; evaluation remains on the original fine labels.

Family logits and posterior. For $p_y(x) = \mathrm{softmax}(s(x))_y$, the aggregation and its normalized distribution are

$$\mathrm{Agg}(s(x))_b = \log \sum_{y \in \mathcal{I}_b} \exp(s_y(x)), \tag{27}$$

$$p_b^{\mathrm{F}}(x) = \mathrm{softmax}(\mathrm{Agg}(s(x)))_b = \sum_{y \in \mathcal{I}_b} p_y(x). \tag{28}$$

The summation marginalizes over fine labels while remaining conditioned on the image. The maximizing family of p^{F} need not equal the family of the single highest-scoring fine class. Consequently, accuracy of aggregated family prediction and correctness of $g(\hat{y})$ are distinct diagnostics.

Temperature convention. The consistency loss uses $\mathrm{softmax}(\mathrm{Agg}(s)/T)$: aggregation is performed first, followed by family-level temperature scaling. For $T \neq 1$, this is generally different from $\mathrm{softmax}(\mathrm{Agg}(s/T))$, which would aggregate a temperature-scaled fine posterior. The former is the convention in Eq. (11); the probability identity above applies directly at unit temperature. Stop-gradient blocks the consistency gradient through the basic-logit target, but the shared representation can still change through other differentiable paths.

Centering, masking, and local indexing. For $y \in \mathcal{I}_b$, $\pi_b(y)$ is its local position in the fixed ordering of that sibling set. The correction is

$$\tilde{r}_y(x; b) = \mathbb{I}[y \in \mathcal{I}_b] \left(r_y(x) - \frac{1}{|\mathcal{I}_b|} \sum_{j \in \mathcal{I}_b} r_j(x) \right). \tag{29}$$

Hence $\sum_{y \in \mathcal{I}_b} \tilde{r}_y = 0$, and outside-family logits receive zero correction. Within-family softmax is invariant to the subtracted common mean, so centering does not change the sibling cross-entropy. Its role is to remove a common family offset during additive merging. Zero-mean logits do not imply preservation of the family's softmax probability mass: the exponential normalization can still change. For a singleton family, both the centered correction and the one-class sibling cross-entropy are zero.

B.2 Commitment Events and Selective Refinement Ratios

The frozen teacher uses the target-aware event in Eq. (16), while inference uses the student's label-free event in Eq. (22). For reference,

$$C_\tau^{\mathrm{train}}(x_i) = \mathbb{I}\left[\mathrm{softmax}(u^{\mathrm{tea}}(x_i))_{b_i} \geq \tau \ \wedge \ \arg\max_b \mathrm{Agg}(s^{\mathrm{tea}}(x_i))_b = b_i \right], \tag{30}$$

$$m_i^{\mathrm{train}} = C_\tau^{\mathrm{train}}(x_i) \mathbb{I}[y_i \in \mathcal{Y}_{\mathrm{mf}}], \tag{31}$$

$$\rho_\tau^{\mathrm{train}} = \frac{1}{N} \sum_{i=1}^{N} m_i^{\mathrm{train}}. \tag{32}$$

The frozen teacher provides a fixed predictor, not a guarantee that every transformed view of an example has the same commitment. Selection can depend on the augmented input. Teacher-based and student-side selection are compared in Appendix F.7.

With $\hat{b}_u = \arg\max_b u_b^{\mathrm{stu}}$, $\hat{b}_s = \arg\max_b \mathrm{Agg}(s_{\mathrm{base}})_b$, and $\hat{y}_{\mathrm{base}} = \arg\max_y s_{\mathrm{base},y}$, inference uses

$$C_{\tau_{\mathrm{eval}}}^{\mathrm{eval}}(x) = \mathbb{I}\left[\max_b \mathrm{softmax}(u^{\mathrm{stu}}(x))_b \geq \tau_{\mathrm{eval}} \ \wedge \ \hat{b}_u(x) = \hat{b}_s(x) \right], \tag{33}$$

$$m^{\mathrm{eval}}(x) = C_{\tau_{\mathrm{eval}}}^{\mathrm{eval}}(x) \mathbb{I}[\hat{y}_{\mathrm{base}}(x) \in \mathcal{Y}_{\mathrm{mf}}], \tag{34}$$

$$s^{\mathrm{final}}(x) = s_{\mathrm{base}}(x) + \alpha_{\mathrm{r}} m^{\mathrm{eval}}(x) \tilde{r}(x; \hat{b}_u(x)), \tag{35}$$

$$\rho_{\tau_{\mathrm{eval}}}^{\mathrm{eval}} = \mathbb{E}_{x \sim \mathcal{P}_{\mathrm{eval}}}[m^{\mathrm{eval}}(x)]. \tag{36}$$

The base predictor is returned exactly when the mask is zero. Training selection uses the true class frequency; evaluation selection uses the base-predicted frequency. Reported shot-wise accuracy is grouped by the true class. These are different partitions, as analyzed in Appendix H.6. The selection ratio describes application of the residual correction, not measured savings from skipping its forward computation.

B.3 Hierarchical Decomposition of Fine-Grained Prediction

The staged design of BASICLT is motivated by the observation that fine-grained recognition can be decomposed into family-level abstraction and within-family differentiation. We formalize this observation with a simple probabilistic identity.

Proposition B.1 (Hierarchical decomposition)**.** *Let $y \in \mathcal{Y}$ be a fine-grained class and let $b = g(y) \in \mathcal{B}$ be its corresponding basic-level family. For any input x,*

$$p(y \mid x) = p(b \mid x)\, p(y \mid b, x). \tag{37}$$

Equivalently,

$$-\log p(y \mid x) = -\log p(b \mid x) - \log p(y \mid b, x). \tag{38}$$

Proof. Because $b = g(y)$, the event $Y = y$ implies $B = b$. Therefore $p(y, b \mid x) = p(y \mid x) = p(b \mid x)p(y \mid b, x)$. Taking negative logarithms gives Eq. (38) when the probabilities are positive; zero-probability cases are interpreted with extended log loss. □

This proposition is mathematically elementary, but it is useful for interpreting BASICLT. It makes explicit that fine-grained prediction contains two semantic components: a family-level component $p(b \mid x)$ and a within-family component $p(y \mid b, x)$. These two components need not have the same level of reliability under long-tailed scarcity. Family-level abstraction can be supported by examples from related fine-grained classes within the same family, whereas sibling-level differentiation may require finer visual evidence and can be more fragile for medium-shot and few-shot classes.

The decomposition therefore motivates the abstraction-to-differentiation design. Stage 1 improves the family-level component by learning an explicit basic-level branch and aligning the fine-grained predictor with the induced family-level distribution. Stage 2 then targets the within-family component through residual refinement, but only when family-level commitment is reliable. In this sense, BASICLT does not assume that coarse prediction should replace fine-grained recognition. Instead, it uses coarse semantic reliability to decide when additional fine-grained differentiation is appropriate.

This interpretation also explains why uniform refinement can be suboptimal. If the family-level component is unreliable, then within-family correction may be applied around the wrong semantic family and can damage the prediction. By contrast, commitment-guided refinement activates the residual branch only when the model has sufficient evidence for the coarse family. The method therefore treats fine-grained refinement as a conditional correction built on top of learned abstraction.

B.4 Conditional Utility of Selective Refinement

We next provide a simple risk decomposition to clarify when selective refinement is preferable to uniform refinement. The purpose of this analysis is not to prove that the proposed commitment rule is universally optimal, but to formalize the design intuition that residual refinement should be applied only on samples where it is expected to be useful.

Let $\ell_{\mathrm{base}}(x, y)$ denote the loss of the base fine-grained prediction before residual correction, and let $\ell_{\mathrm{ref}}(x, y)$ denote the loss after applying residual refinement. Define the refinement advantage as

$$\Delta(x, y) = \ell_{\mathrm{ref}}(x, y) - \ell_{\mathrm{base}}(x, y). \tag{39}$$

A negative value of $\Delta(x, y)$ indicates that refinement improves the prediction on sample (x, y), while a positive value indicates that refinement is harmful.

For a commitment event $C(x) \in \{0, 1\}$, the selectively refined loss is

$$\ell_{\mathrm{sel}}(x, y) = C(x)\, \ell_{\mathrm{ref}}(x, y) + \big(1 - C(x)\big)\, \ell_{\mathrm{base}}(x, y). \tag{40}$$

In the context of BASICLT, $C(x)$ can be understood as the evaluation-time commitment event together with the low-frequency eligibility mask. For simplicity, we write it as a single binary variable.

Proposition B.2 (Risk comparison between selective and uniform refinement)**.** *For $0 < \mathbb{P}(C = 1) < 1$, assume that refinement is no worse than the base predictor on committed samples in expectation,*

$$\mathbb{E}[\Delta(x, y) \mid C(x) = 1] \le 0, \tag{41}$$

and no better than the base predictor on uncommitted samples in expectation,

$$\mathbb{E}[\Delta(x,y) \mid C(x)=0] \geq 0. \tag{42}$$

Then the expected risk of selective refinement is no larger than that of uniform refinement:

$$\mathbb{E}\big[\ell_{\mathrm{sel}}(x,y)\big] \leq \mathbb{E}\big[\ell_{\mathrm{ref}}(x,y)\big]. \tag{43}$$

Proof. Let

$$\mathcal{R}_{\mathrm{base}} = \mathbb{E}\big[\ell_{\mathrm{base}}(x,y)\big], \qquad \mathcal{R}_{\mathrm{ref}} = \mathbb{E}\big[\ell_{\mathrm{ref}}(x,y)\big], \tag{44}$$

and

$$\mathcal{R}_{\mathrm{sel}} = \mathbb{E}\big[\ell_{\mathrm{sel}}(x,y)\big]. \tag{45}$$

Using Eq. (39), the uniformly refined risk can be written as

$$\mathcal{R}_{\mathrm{ref}} = \mathcal{R}_{\mathrm{base}} + \mathbb{E}[\Delta(x,y)]. \tag{46}$$

Similarly, from Eq. (40), the selective risk is

$$\mathcal{R}_{\mathrm{sel}} = \mathbb{E}[C(x)\ell_{\mathrm{ref}}(x,y) + (1-C(x))\ell_{\mathrm{base}}(x,y)] \tag{47}$$

$$= \mathcal{R}_{\mathrm{base}} + \mathbb{E}[C(x)\Delta(x,y)]\,. \tag{48}$$

By the law of total expectation,

$$\mathbb{E}[\Delta(x,y)] = \mathbb{P}(C=1)\mathbb{E}[\Delta(x,y) \mid C=1] + \mathbb{P}(C=0)\mathbb{E}[\Delta(x,y) \mid C=0], \tag{49}$$

$$\mathbb{E}[C(x)\Delta(x,y)] = \mathbb{P}(C=1)\mathbb{E}[\Delta(x,y) \mid C=1]. \tag{50}$$

Therefore,

$$\mathcal{R}_{\mathrm{ref}} - \mathcal{R}_{\mathrm{sel}} = \mathbb{P}(C=0)\mathbb{E}[\Delta(x,y) \mid C=0]. \tag{51}$$

Under assumption Eq. (42), the right-hand side is nonnegative. Hence,

$$\mathcal{R}_{\mathrm{sel}} \leq \mathcal{R}_{\mathrm{ref}}. \tag{52}$$

This proves the claim. □

Proposition B.2 formalizes a simple condition under which selective refinement is preferable to applying refinement uniformly. If refinement is useful on committed samples but harmful or unnecessary on uncommitted samples, then applying refinement everywhere introduces avoidable risk. A selective policy can avoid this risk by falling back to the base prediction on uncommitted samples.

A companion comparison explains when selective refinement improves over the base predictor.

Remark B.1 (When selective refinement improves over the base predictor). From Eq. (48), we have

$$\mathcal{R}_{\mathrm{sel}} = \mathcal{R}_{\mathrm{base}} + \mathbb{P}(C=1)\mathbb{E}[\Delta(x,y) \mid C=1]. \tag{53}$$

Therefore, if

$$\mathbb{E}[\Delta(x,y) \mid C=1] < 0 \tag{54}$$

and

$$\mathbb{P}(C=1) > 0, \tag{55}$$

then

$$\mathcal{R}_{\mathrm{sel}} < \mathcal{R}_{\mathrm{base}}. \tag{56}$$

Thus, selective refinement improves over the base predictor whenever the committed subset is non-empty and refinement is beneficial on that subset in expectation.

When selection is identically zero or one, the corresponding predictor is exactly the base or uniformly refined predictor. The conditional assumptions are statements about expected loss on the selected and unselected subsets; confidence and agreement do not establish them automatically.

This analysis highlights the role of the commitment event. The commitment rule does not need to perfectly identify every sample on which refinement is useful. It only needs to separate, on average, a subset where refinement has higher utility from a subset where refinement is less reliable. This is consistent with the empirical threshold behavior reported in the main text: overly permissive thresholds reduce reliability, while overly strict thresholds reduce coverage.

In BASICLT, the commitment event is designed to capture semantic reliability at the family level. If the family-level prediction is unreliable, then a within-family residual correction may be applied around the wrong semantic family and can degrade the prediction. If the family-level prediction is reliable, then sibling-level correction is more likely to refine the decision among appropriate candidate classes. This provides the rationale for using family-level commitment as the control signal for residual refinement.

B.5 Selective Complexity Control

The selective refinement ratio provides another perspective on BASICLT. Uniform refinement increases the effective fine-grained capacity for all samples, regardless of whether additional sibling-level discrimination is needed or reliable. In contrast, BASICLT activates the residual correction only on a committed subset of samples. The refinement ratio therefore measures not only how often the residual branch is used, but also how aggressively the model allocates additional fine-grained capacity.

Let

$$\rho_{\tau_{\mathrm{eval}}}^{\mathrm{eval}} = \mathbb{E}_{x\sim\mathcal{P}_{\mathrm{eval}}}\left[m^{\mathrm{eval}}(x)\right] \tag{57}$$

be the evaluation-time selective refinement ratio defined in Eq. (36). A small $\rho_{\tau_{\mathrm{eval}}}^{\mathrm{eval}}$ means that the model mostly relies on the base fine-grained prediction and activates residual correction only conservatively. A large $\rho_{\tau_{\mathrm{eval}}}^{\mathrm{eval}}$ means that residual refinement is applied more broadly.

Coverage–reliability trade-off. The commitment threshold controls a trade-off between coverage and reliability. A lower threshold increases the refinement ratio by allowing more samples to receive residual correction. However, it may also include samples whose family-level predictions are unreliable, making the residual correction more likely to operate within an incorrect family. A higher threshold improves the reliability of the selected subset but may miss samples that could benefit from refinement. This explains why performance can exhibit an intermediate optimum as the evaluation threshold varies.

Conditional capacity allocation. From the perspective of model capacity, the residual branch provides additional sibling-level discrimination. Applying this capacity uniformly can be unnecessary or harmful, especially when the model has not formed reliable coarse semantic structure. Selective refinement instead allocates this additional capacity conditionally. The base predictor handles all samples, while the residual branch is activated only when the family-level commitment criterion is satisfied and the base prediction lies in the target low-frequency set.

This conditional allocation is especially appropriate for long-tailed recognition. Head classes often have enough examples to support direct fine-grained discrimination, while medium-shot and few-shot classes can benefit more from additional structure. By restricting residual refinement to committed low-frequency predictions, BASICLT avoids imposing extra fine-grained corrections universally and focuses the additional capacity on the regimes where it is most needed.

Relation to abstraction-to-differentiation. Selective complexity control is closely tied to the abstraction-to-differentiation principle. Stage 1 learns a basic-level abstraction layer that provides coarse semantic reliability. Stage 2 uses this reliability to decide whether to introduce additional sibling-level differentiation. Thus, the model does not merely add another classifier or prediction head. Instead, it increases discriminative complexity only after the corresponding coarse semantic structure has become reliable.

The centered residual formulation in Eq. (29) further reinforces this interpretation. Because the residual is centered and masked within the committed family, it modifies relative scores among sibling classes without changing scores outside that family. The residual branch therefore acts as a local correction mechanism rather than as an unconstrained global classifier.

Limitations of the formal view. The analysis above relies on expected refinement utility and does not imply that the commitment rule is optimal for every sample. In practice, the commitment event is a heuristic but useful proxy for refinement reliability. It may fail when the basic-level branch is overconfident, when the taxonomy groups visually dissimilar classes, or when the base prediction falls into an incorrect frequency group at inference time. These cases motivate the empirical analyses on threshold sensitivity, taxonomy construction, and failure modes reported in later appendices.

Overall, the formal view supports the central design of BASICLT. Long-tailed recognition can benefit not only from changing the amount of supervision assigned to different classes, but also from controlling the semantic granularity at which discrimination is imposed. By learning basic-level abstraction first and activating sibling-level residual refinement selectively, BASICLT provides a practical mechanism for conditional fine-grained differentiation under scarce and imbalanced supervision.

Table 5: Summary of the basic-level taxonomies used by BASICLT. Each mapping is fixed before training and is used only for abstraction learning and selective refinement. Final evaluation remains on the original fine-grained label space.

Dataset	# Fine classes	# Basic families	Taxonomy source	Construction type
CIFAR-100-LT	100	20	Official CIFAR-100 superclasses	Official hierarchy
CIFAR-10-LT	10	2	WordNet coarse semantics	Fixed semantic grouping
ImageNet-LT	1000	107	ImageNet / WordNet hierarchy	Fixed curated hierarchy
iNaturalist 2018	8142	1120	Public biological taxonomy	Fixed public hierarchy
Places-LT	365	16	Scene-level semantics	Fixed semantic grouping with audit

C Dataset-Specific Basic Taxonomy Construction

This appendix describes how the fixed basic-level taxonomy is constructed for each benchmark used in our experiments. Since BASICLT relies on a mapping from fine-grained classes to basic-level families, the construction of this mapping is important for both reproducibility and interpretation. The purpose of the taxonomy is not to redefine the recognition task, but to provide an intermediate semantic scaffold for Stage 1 abstraction learning and Stage 2 selective within-family refinement. All reported results are evaluated on the original fine-grained label space.

Across all datasets, the taxonomy construction follows three principles.

1. **Fixed before training.** The mapping from fine-grained classes to basic-level families is constructed before model training and is not optimized jointly with the model.
2. **Semantically meaningful.** Classes assigned to the same family should share a coarse semantic relationship, such as belonging to the same superclass, biological group, object family, or scene type.
3. **Moderate granularity.** The basic-level taxonomy should be coarser than the fine-grained label space but not so coarse that within-family refinement becomes meaningless. The goal is to provide a stable abstraction layer while preserving non-trivial sibling-level differentiation.

Table 5 summarizes the taxonomies used in the main experiments. For datasets with small or official taxonomies, the mapping is listed directly in this appendix. For large-scale datasets, the complete fine-to-basic mapping is included in the supplementary taxonomy files used by the training code.

C.1 CIFAR-100-LT Basic Taxonomy

CIFAR-100 contains 100 fine-grained classes grouped into 20 official superclasses, with each superclass containing exactly 5 fine-grained classes. For CIFAR-100-LT, we use these official superclass annotations as the basic-level taxonomy. This gives

$$K = 100, \qquad M = 20, \qquad |\mathcal{I}_b| = 5 \quad \text{for all } b \in \mathcal{B}. \tag{58}$$

This benchmark provides the cleanest setting for evaluating the proposed abstraction-to-differentiation mechanism, because the basic-level taxonomy is official, balanced, and independent of our method. No manual modification is applied to the CIFAR-100 superclass structure.

Using the official superclass taxonomy has two advantages. First, it avoids introducing additional design choices into the benchmark. Second, it provides a semantically meaningful family structure for studying whether rare classes benefit from first learning coarse abstraction and then performing sibling-level refinement. Therefore, CIFAR-100-LT is used as the primary controlled benchmark for ablation and mechanism analysis in the main paper.

C.2 CIFAR-10-LT Basic Taxonomy

CIFAR-10 does not provide an official superclass taxonomy. We therefore use a fixed coarse semantic taxonomy constructed before training from WordNet-derived semantic relations. Specifically, each CIFAR-10 class name is associated with a corresponding WordNet synset when available, and the classes are grouped according to high-level semantic relations. Since CIFAR-10 contains only 10

Table 6: Fixed CIFAR-10-LT basic taxonomy. The grouping is constructed before training from coarse semantic relations and is not selected using validation or test accuracy.

Family ID	Basic family	Fine-grained classes
0	Artifact / vehicle	`airplane`, `automobile`, `ship`, `truck`
1	Living organism	`bird`, `cat`, `deer`, `dog`, `frog`, `horse`

classes, the resulting taxonomy is intentionally shallow and provides a lightweight abstraction layer rather than a deep hierarchy.

Table 6 summarizes the fixed two-family taxonomy used for CIFAR-10-LT. Family 0 contains artifacts and vehicles, while family 1 contains living organisms. Thus, the taxonomy has

$$K = 10, \qquad M = 2. \tag{59}$$

This taxonomy is fixed before training and is not selected using validation or test accuracy. The implementation records the standard CIFAR-10 class order, the fine-to-family mapping, the family names, and the semantic references used for construction. During data loading, the class order is checked against the standard CIFAR-10 order to prevent accidental class-family misalignment.

We use this shallow taxonomy because forcing a finer hierarchy on CIFAR-10 would make the basic-level task nearly identical to the original fine-grained task, while using a single family would remove the abstraction signal entirely. The two-family taxonomy therefore provides the most meaningful coarse semantic partition at this dataset scale.

Although this taxonomy is simpler than the official CIFAR-100 superclass taxonomy, it provides a useful robustness check. It tests whether the proposed framework can still operate when the available hierarchy is shallow and coarse. The complete fine-to-family mapping is fixed across all CIFAR-10-LT experiments.

C.3 ImageNet-LT Basic Taxonomy

ImageNet-LT contains 1000 fine-grained categories, each associated with an ImageNet synset in the WordNet hierarchy. Unlike CIFAR-100, however, ImageNet does not provide an official balanced superclass taxonomy. A naive WordNet depth cut can easily produce overly generic families, such as `artifact.n.01`, `object.n.01`, or `entity.n.01`, which are too broad to serve as useful basic-level abstractions.

We therefore construct a fixed curated WordNet family taxonomy for ImageNet-LT. The goal is to obtain a family structure that is coarser than the original 1000-way label space, semantically coherent, and still useful for within-family differentiation. The taxonomy is constructed once before training and is not selected using validation or test accuracy.

The construction proceeds in three steps.

Candidate ancestor extraction. For each ImageNet class, we collect candidate ancestors from its WordNet hypernym paths. Ancestors that are too generic, too close to the root, or visually uninformative are excluded. This prevents the taxonomy from collapsing into broad categories that provide little useful abstraction.

Curated family-anchor selection. We select a fixed set of WordNet anchors that function as basic-level families. The anchor set is designed to avoid overly broad groups while preserving semantic coherence. When a high-level ancestor would create a very large and heterogeneous family, we replace it with more specific family-like anchors. For example, broad groups involving dogs, birds, garments, vehicles, and devices are split into more coherent subfamilies when WordNet provides suitable intermediate ancestors. This produces a more controlled family structure than a naive WordNet depth cut.

Coverage audit and deterministic correction. After assigning fine-grained classes to anchors, we verify that every ImageNet-LT class is covered by exactly one family. Classes whose hypernym

Table 7: Summary of the fixed ImageNet-LT curated WordNet family taxonomy. The taxonomy is constructed before training and is not selected using validation or test accuracy.

Property	Value
Source hierarchy	ImageNet / WordNet
Construction type	Fixed curated semantic hierarchy
Anchor selection	Curated WordNet family anchors
Performance-based tuning	No
Fine-grained label space	1000 classes
Basic-level family space	107 families
Fine-class coverage	1000 / 1000
Assignment rule	One family per fine class
Family index validity	Contiguous family indices
Singleton families	8
Family-size range	1–33 classes
Fine classes per family	9.35 ± 8.42
Training images per family	1082.67 ± 1022.49
Training-image range per family	7–4401
Large-family audit flags	24 families

Table 8: Largest ImageNet-LT families in the curated WordNet taxonomy. These groups are reported for semantic auditing and transparency; the audit flags are not used for model training or selection.

ID	WordNet family anchor	# Fine classes	# Train images	Audit flag
53	`implement.n.01`	33	2735	large family, many images
28	`diapsid.n.01`	31	3460	large family, many images
105	`working_dog.n.01`	30	2369	large family, many images
55	`instrument.n.01`	28	3283	large family, many images
54	`insect.n.01`	27	3435	large family, many images
96	`terrier.n.01`	26	2176	large family, many images
47	`garment.n.01`	26	3471	large family, many images
64	`musical_instrument.n.01`	26	4401	large family, many images
27	`device.n.01`	26	2178	large family, many images
80	`protective_covering.n.01`	25	2965	large family, many images
02	`aquatic_bird.n.01`	24	3001	large family, many images
20	`container.n.01`	23	1711	large family
79	`produce.n.01`	23	2355	large family, many images
63	`motor_vehicle.n.01`	22	2426	large family, many images
102	`vessel.n.03`	22	2748	large family, many images

paths do not match a selected anchor are resolved by deterministic coverage corrections or by adding a suitable family-level anchor. These corrections are applied before training and are based only on taxonomy validity and semantic inspection, not on model performance.

Table 7 summarizes the final taxonomy. The mapping covers all 1000 fine-grained classes, contains 107 basic-level families, and assigns each class to exactly one family. The observed family size ranges from 1 to 33 classes.

Table 8 lists the largest families identified during semantic auditing. These families are shown to make the taxonomy construction transparent. The audit flags indicate relatively large family size and/or many training images, but they are not used for model training, model selection, or taxonomy tuning.

The largest groups remain semantically interpretable, including insects, terriers, musical instruments, motor vehicles, vessels, aquatic birds, and working dogs. Some broader anchors, such as `implement.n.01`, `device.n.01`, and `protective_covering.n.01`, are retained because further splitting them using WordNet would either create unstable singleton-heavy groups or require less

reproducible manual decisions. Importantly, the maximum family size is bounded at 33 fine-grained classes, avoiding the excessively broad groups produced by naive WordNet cuts.

We intentionally do not force all families to have equal size. Equal-size grouping may improve numerical regularity but can break semantic coherence. Instead, we prioritize semantically meaningful WordNet families and allow natural variation in family size. Singleton families are retained when a class does not have a suitable nearby sibling group under the selected granularity. For such classes, family-level abstraction remains well defined, while within-family residual refinement is naturally limited. Larger families provide richer sibling-level differentiation and are particularly useful for the refinement stage.

This curated taxonomy is fixed across all ImageNet-LT experiments. The complete fine-to-family mapping is included in the supplementary taxonomy files. Appendix F.16 reports taxonomy sensitivity results, including comparisons with coarser, finer, random, and shuffled groupings. These analyses verify that the gain is not merely caused by adding an auxiliary branch or reducing the label space, but depends on a semantically meaningful family structure.

C.4 iNaturalist 2018 Basic Taxonomy

iNaturalist 2018 is a large-scale fine-grained recognition benchmark with a natural biological hierarchy. Each category belongs to a taxonomic structure that includes levels such as kingdom, phylum, class, order, family, genus, and species. Since the original recognition task is species-level classification, this hierarchy provides a natural source of intermediate abstraction.

For iNaturalist 2018, we derive the basic-level taxonomy from the public biological annotations associated with the dataset. The fine-grained species labels are mapped to a fixed intermediate biological level. In our main experiments, we use a family-level biological grouping when available. If a class lacks a valid family-level annotation in the metadata, we assign it using the nearest available valid ancestor under the same public taxonomy. The resulting mapping is fixed before training and is used consistently across all experiments.

This construction is well aligned with the motivation of BASICLT. Many iNaturalist categories are visually similar species that share higher-level biological structure. For such data, family-level abstraction can provide shared support across related rare species, while Stage 2 residual refinement can focus on sibling-level differences when the family prediction is reliable.

The complete iNaturalist fine-to-basic mapping is included in the supplementary taxonomy file. For reproducibility, the file records the original species label, its biological ancestors, and the final basic-level family index used by the model. The mapping is not tuned using validation accuracy.

C.5 Places-LT Basic Taxonomy

Places-LT contains 365 scene categories and does not provide a single official basic-level taxonomy aligned with our training objective. We therefore construct a fixed coarse scene taxonomy before training. The goal is to evaluate whether BASICLT remains useful beyond object-centric and biological hierarchies.

We group the 365 scene categories into 16 broad scene families. The families correspond to coarse scene semantics such as indoor residential scenes, indoor public scenes, commercial scenes, educational and cultural scenes, industrial scenes, transportation-related scenes, urban outdoor scenes, rural outdoor scenes, natural landscapes, water-related scenes, sports and recreation scenes, religious or ceremonial scenes, food-related scenes, workspaces, entertainment scenes, and miscellaneous structured environments. The exact mapping from each Places-LT class to its scene family is provided in the supplementary taxonomy file and is fixed across all experiments.

Table 9 reports the audited Places-LT scene-family taxonomy used in our experiments. The taxonomy contains 16 non-empty families, with family sizes ranging from 4 to 35 fine-grained classes. The distribution is intentionally not forced to be uniform, because semantic coherence is prioritized over numerical balance. The final mapping remains fixed across all Places-LT experiments and is used only to provide the intermediate abstraction signal; all reported accuracies are evaluated on the original 365-way label space.

Table 9: Detailed statistics of the fixed Places-LT scene-family taxonomy. The mapping is constructed before training by semantic auditing and is not tuned using validation or test accuracy. "Avg." denotes the average number of training images per fine-grained class within each family.

ID	Scene family	#Cls.	Cls. %	#Train	Train %	Avg.	Representative classes
00	Indoor residential	32	8.8	5622	8.3	175.7	bedroom, bathroom, living_room, house, patio
01	Indoor public	21	5.8	2203	3.3	104.9	corridor, lobby, hotel_room, hospital_room, staircase
02	Education / culture	18	4.9	2595	3.8	144.2	art_gallery, classroom, library_indoor, museum_indoor, lecture_room
03	Industrial / workplace	21	5.8	1528	2.3	72.8	assembly_line, auto_factory, clean_room, office, construction_site
04	Transportation	28	7.7	3348	5.0	119.6	airfield, airport_terminal, highway, parking_lot, train_station_platform
05	Urban built environment	19	5.2	5892	8.7	310.1	alley, arch, downtown, skyscraper, street
06	Rural / agricultural	23	6.3	2205	3.3	95.9	barn, farm, field_cultivated, pasture, vineyard
07	Natural landscape	31	8.5	6400	9.5	206.5	canyon, forest_path, mountain, rainforest, volcano
08	Water / coastal	31	8.5	10325	15.3	333.1	beach, coast, harbor, lake_natural, waterfall
09	Sports / recreation	35	9.6	11333	16.8	323.8	amusement_park, baseball_field, ski_slope, stadium_football, swimming_pool_outdoor
10	Religious / historic	15	4.1	3860	5.7	257.3	castle, cemetery, church_indoor, mosque_outdoor, temple_asia
11	Food / dining	24	6.6	2311	3.4	96.3	bakery_shop, bar, dining_room, restaurant, kitchen
12	Entertainment / leisure	13	3.6	638	0.9	49.1	amphitheater, arcade, ballroom, discotheque, stage_indoor
13	Shopping / service	35	9.6	1194	1.8	34.1	bookstore, department_store, market_outdoor, supermarket, beauty_salon
14	Infrastructure / utility	15	4.1	2835	4.2	189.0	bridge, dam, lighthouse, tower, water_tower
15	Miscellaneous structured	4	1.1	211	0.3	52.8	pavilion, sky, storage_room, trench
Total	–	365	100.0	62500	100.0	184.9	–

This taxonomy is necessarily less canonical than the CIFAR-100 superclass taxonomy or the iNaturalist biological hierarchy. Nevertheless, it provides a meaningful test of the proposed framework. If BASICLT only worked with clean object or biological hierarchies, its benefit would be limited on Places-LT. The observed improvement on Places-LT suggests that abstraction-to-differentiation can also be useful when the taxonomy captures broader scene-level semantics.

C.6 Taxonomy Sensitivity Protocol

Because BASICLT relies on a fixed basic-level taxonomy, we evaluate how sensitive the method is to the quality and granularity of this taxonomy. The main sensitivity analysis is conducted on CIFAR-100-LT, where the official superclass taxonomy provides a clean reference point. The large-scale taxonomies are summarized separately through their construction and family-distribution statistics.

We consider the following variants.

Official or semantic taxonomy. This is the main taxonomy used in the reported experiments. For CIFAR-100-LT, it corresponds to the official superclass labels. For ImageNet-LT, it is derived from a fixed curated WordNet family taxonomy; for iNaturalist 2018, it is derived from the public biological hierarchy.

Coarser taxonomy. We merge related basic-level families into larger groups. This reduces the number of basic families and makes the abstraction task easier, but it also makes each family more semantically diverse. Coarser taxonomies test whether overly broad abstraction weakens within-family refinement.

Finer taxonomy. We split some families into smaller subgroups. This increases the number of basic families and makes the abstraction layer closer to the original fine-grained task. Finer taxonomies test whether insufficient abstraction reduces the benefit of Stage 1.

Random grouping. Fine-grained classes are assigned to random families while preserving the same number of families as the main taxonomy. This variant tests whether the gain is merely caused by adding an auxiliary branch with fewer labels.

Shuffled semantic grouping. We preserve the family-size distribution of the semantic taxonomy but shuffle class assignments across families. This variant controls for group-size statistics while removing semantic coherence.

Noisy taxonomy. A fraction of fine-grained classes is randomly reassigned to incorrect families. This variant evaluates robustness to imperfect taxonomy construction.

The sensitivity results show that semantically meaningful taxonomies perform best. Coarser and finer variants reduce performance in different ways: overly coarse taxonomies weaken sibling-level refinement, while overly fine taxonomies weaken the abstraction effect. Random and shuffled taxonomies perform worse than semantic taxonomies, indicating that the improvement is not simply due

to adding another prediction head or increasing supervision. These results support the interpretation that the family structure used by BASICLT provides meaningful semantic abstraction.

C.7 Discussion

The taxonomies used in our experiments vary in how canonical they are. CIFAR-100-LT uses an official and balanced superclass taxonomy. ImageNet-LT and iNaturalist 2018 derive basic-level families from public semantic or biological hierarchies. CIFAR-10-LT and Places-LT use fixed semantic groupings defined before training. This variation allows us to evaluate BASICLT under both clean hierarchical settings and less canonical semantic groupings.

At the same time, the dependence on a fixed taxonomy is a limitation of the current framework. If the taxonomy groups visually dissimilar classes together, the basic-level abstraction may become less useful. If the taxonomy is too coarse, within-family refinement becomes difficult because each family contains overly diverse classes. If the taxonomy is too fine, the abstraction task becomes too similar to the original fine-grained task and provides limited shared support. These limitations motivate future work on automatically discovering task-adaptive basic-level structure or learning soft taxonomies that better reflect visual similarity under long-tailed data.

In this work, we use fixed taxonomies to keep the framework transparent and reproducible. The sensitivity analysis shows that taxonomy quality matters, but also that the method does not require a perfectly balanced or manually optimized hierarchy. The key requirement is that the basic-level mapping provides a meaningful intermediate semantic structure that can support abstraction before differentiation.

C.8 Single Membership and Imbalance among Families

A fixed partition. The implemented taxonomy assigns each fine class to one family. This makes aggregation, sibling indexing, commitment, and one-family residual merging unambiguous. Multi-inheritance is not part of the evaluated method. A possible extension would use a membership matrix $A \in [0,1]^{K\times M}$ with $\sum_b A_{yb} = 1$, inducing $p_b^{\mathrm{F}}(x) = \sum_y A_{yb} p_y(x)$. However, overlapping memberships would also require a new rule for selecting and combining family-specific corrections. Separate centering of overlapping corrections does not by itself resolve their interactions. Such extensions remain unevaluated.

Unequal family size and frequency. Equal numbers of fine classes per family do not imply equal family training counts. Let $N_b = \sum_{y\in\mathcal{I}_b} n_y$. The ImageNet-LT mapping contains 1–33 fine classes and 7–4401 training images per family; the Places-LT mapping contains 4–35 classes and 211–11333 images per family. The framework therefore does not assume a balanced coarse-label distribution. The cross-benchmark comparison in Appendix F.15 shows the reported selective-refinement gains under these nonuniform mappings, but does not isolate a causal effect of family imbalance. Highly sparse or visually heterogeneous families can still weaken both abstraction and commitment.

Construction when an official hierarchy is unavailable. A practical construction procedure is to form an intermediate partition from an ontology, metadata, domain expertise, or auditable visual clustering before training; verify complete, single-family coverage; inspect oversized heterogeneous groups; and keep the mapping fixed for comparisons. This does not establish that an effective hierarchy can be obtained automatically for every dataset.

The fixed-seed comparison supports coherent visual grouping as a possible alternative to semantic grouping in this setting. The multi-run perturbation study separately measures degradation under random, shuffled, noisy, coarser, and finer mappings. Neither experiment proves that taxonomy choice is inconsequential. Family-balanced objectives, per-family thresholds, and learned soft taxonomies are possible extensions, not components of the reported pipeline.

D Extended Implementation Details

This appendix provides implementation details for BASICLT. The purpose is to make the two-stage training and inference pipeline reproducible and to clarify how the abstract formulation in Section 3

Table 10: Fixed-seed taxonomy-construction analysis on CIFAR-100-LT with IR=100. The entries are single-seed Top-1 accuracies (%), not means over five seeds. They are distinct from the multi-run perturbation study in Table 30.

Construction	Overall accuracy
Semantic grouping	60.6
Visual clustering	60.1
Frequency grouping	59.2
Random balanced grouping	58.1
Random unbalanced grouping	57.6

Table 11: CIFAR model instantiation used in the main controlled experiments.

Item	Value
Backbone family	ResNet-32 style
Prediction branches	3
Fine-grained output aggregation	Mean of branch logits
Basic branch input	Mean of branch features
Basic-level output dimension on CIFAR-100-LT	20
Residual output dimension on CIFAR-100-LT	100
Residual merge mode	Additive
Residual centering	Enabled
Residual family masking	Enabled

is instantiated in experiments. We describe the model components, Stage 1 and Stage 2 training procedures, commitment thresholds, selective residual inference, and reproducibility settings.

D.1 Model Components and Training Configuration

BASICLT is implemented by augmenting a fine-grained model with two additional components: a basic-level branch and a family-conditioned residual refinement branch. The original fine-grained predictor remains responsible for the target recognition task over the label space $\mathcal{Y}$. The added branches provide intermediate abstraction and selective differentiation.

Fine-grained predictor. The fine-grained predictor takes an input image x, extracts a visual representation $z = f_\theta(x)$, and produces fine-grained logits

$$s(x) \in \mathbb{R}^K. \tag{60}$$

For CIFAR experiments, we instantiate the predictor using a ResNet-32 style architecture with multiple prediction branches. The branch outputs are aggregated to obtain the fine-grained logits used by the original recognition objective. The same fine-grained predictor architecture and training protocol are used for controlled variants unless otherwise specified.

CIFAR instantiation. For CIFAR experiments, the fine-grained predictor is instantiated as a ResNet-32 style multi-branch model. Let E denote the number of prediction branches. Each branch produces logits $s^{(e)}(x)$, and the fine-grained logits before residual refinement are obtained by branch averaging:

$$s(x) = \frac{1}{E} \sum_{e=1}^{E} s^{(e)}(x). \tag{61}$$

Unless otherwise specified, we use $E = 3$ branches in CIFAR experiments. The same instantiated predictor is used across the base predictor, Stage 1, Stage 2, and all controlled variants.

Basic-level branch. The basic-level branch is attached to the shared representation and produces family logits

$$u(x) \in \mathbb{R}^M. \tag{62}$$

Table 12: Main implementation configuration for CIFAR-100-LT IR$= 100$. This setting is used for the controlled ablations and mechanism analyses unless otherwise specified.

Item	Value
Fine-grained classes	100
Basic-level families	20
Batch size	128
Optimizer	SGD
Momentum	0.9
Weight decay	5×10^{-4}
Nesterov	Yes
Data augmentation	Random crop, horizontal flip, RandAugment
Stage 1 epochs	200
Stage 1 initial learning rate	0.1
Stage 1 LR decay epochs	160, 180
Stage 1 warmup epochs	5
Stage 2 epochs	80
Stage 2 initial learning rate	0.02
Stage 2 LR decay epochs	50, 70
Stage 2 warmup epochs	2
Stage 2 freeze-main epochs	10
Training commitment threshold τ	0.55
Evaluation commitment threshold τ_{eval}	0.60
Target refinement groups	Medium-shot and few-shot
Coarse agreement required	Yes

For CIFAR-100-LT, $M = 20$, corresponding to the official CIFAR-100 superclass taxonomy. When the underlying fine-grained predictor contains multiple branch features, the basic branch uses their mean representation as its input unless otherwise specified. This branch is trained by direct basic-level supervision and is used to provide the abstraction signal for hierarchical consistency and commitment estimation.

Family-conditioned residual branch. Stage 2 introduces a residual refinement branch that produces residual logits

$$r(x) \in \mathbb{R}^K. \tag{63}$$

The residual is interpreted as a within-family correction rather than as a standalone classifier. Given a selected family b, the residual is centered and masked within the sibling set $\mathcal{I}_b$, following Eq. (15). This ensures that the residual branch changes relative scores among sibling classes while leaving classes outside the selected family unchanged.

Optimization. All models are optimized with stochastic gradient descent. For the main CIFAR-100-LT IR$= 100$ setting used in ablations, Stage 1 is trained for 200 epochs with batch size 128, initial learning rate 0.1, momentum 0.9, weight decay 5×10^{-4}, and Nesterov momentum. The learning rate is decayed by a factor of 0.1 at epochs 160 and 180, with 5 warmup epochs. Stage 2 is trained for 80 epochs with initial learning rate 0.02, the same momentum and weight decay, and learning-rate decay at epochs 50 and 70 with 2 warmup epochs.

Data augmentation. For CIFAR experiments, we use standard random crop and horizontal flip together with RandAugment. The same augmentation policy is used across the corresponding controlled variants. For larger-scale datasets, we follow the standard preprocessing and augmentation protocols used for the corresponding benchmark. Dataset-specific configurations are included in the released configuration files.

Main controlled setting. Table 12 summarizes the main CIFAR-100-LT IR$= 100$ configuration used for the ablation and mechanism analysis in the main paper.

Table 13: Dataset-wise training configuration. Epochs and learning rates are shown as Stage 1 / Stage 2. All runs use SGD with momentum 0.9.

Dataset	Backbone	Epochs	Batch size	Initial learning rates	Weight decay
CIFAR-100-LT	3-expert ResNet-32	200 / 80	128	0.1 / 0.02	5×10^{-4}
CIFAR-10-LT	3-expert ResNet-32	200 / 80	128	0.1 / 0.02	5×10^{-4}
ImageNet-LT	3-expert ResNet-50	180 / 15	192	0.075 / 0.001	5×10^{-4}
iNaturalist 2018	3-expert ResNet-50	100 / 15	64	0.035 / 0.001	2×10^{-4}
Places-LT	3-expert ResNet-50	100 / 15	256	0.1 / 0.001	5×10^{-4}

Table 14: Dataset-wise abstraction, refinement, and selection settings. "Scheduled" refers to the dataset configuration; the CIFAR-100-LT schedules are given explicitly in Sections D.2 and D.3. The residual merge coefficient is separate from the sibling-loss weight.

Dataset	$\lambda_{\mathrm{b}}/\lambda_{\mathrm{h}}$	$\eta_{\mathrm{b}}/\eta_{\mathrm{h}}/\lambda_{\mathrm{s}}$	τ	τ_{eval}	α_{r}
CIFAR-100-LT	Scheduled	Scheduled	0.55	0.60	0.06
CIFAR-10-LT	Scheduled	Scheduled	0.65	0.70	0.06
ImageNet-LT	0.012 / 0.004	0.005 / 0.005 / 0.03	0.55	0.60	0.06
iNaturalist 2018	0.010 / 0.005	0.005 / 0.005 / 0.03	0.55	0.60	0.06
Places-LT	0.010 / 0.005	0.005 / 0.005 / 0.03	0.55	0.60	0.06

Dataset-wise configurations. Tables 13 and 14 give the dataset-specific settings. All five datasets use a three-expert fine-grained predictor: ResNet-32 for CIFAR and ResNet-50 for the large-scale datasets. Controlled variants keep the corresponding underlying predictor fixed.

CIFAR uses learning-rate decay at 160/180 epochs in Stage 1 and 50/70 in Stage 2, with 5/2 learning-rate warmup epochs and a 10-epoch Stage 2 main-path freeze. Large-scale runs use cosine decay and a two-epoch residual warmup in Stage 2. Mixup strengths are 0.3, 0.2, and 0.2 for ImageNet-LT, iNaturalist 2018, and Places-LT, respectively. The consistency and distillation temperatures are $T = T_{\mathrm{d}} = 2$.

For large-scale Stage 2, the original main-loss weight is 0.2, while the fine- and basic-distillation weights are 1.0 and 0.05. In the compact objective, $\mathcal{L}_{\mathrm{fine}}$ denotes this configured original fine-grained term: writing the unweighted base objective as $\widehat{\mathcal{L}}_{\mathrm{fine}}$, it is $\mathcal{L}_{\mathrm{fine}} = w_{\mathrm{main}}\widehat{\mathcal{L}}_{\mathrm{fine}}$, with $w_{\mathrm{main}} = 0.2$ for large-scale Stage 2. CIFAR Stage 2 retains the main-loss coefficient 1 and distillation weights 0.5/0.2. The effective coefficient of each residual penalty is 5×10^{-4}, as defined below.

D.2 Stage 1 Training Details

Stage 1 establishes the basic-level abstraction layer. The model is trained from scratch using the original fine-grained recognition objective together with the basic-level supervision loss and the hierarchical consistency loss.

The Stage 1 objective is

$$\mathcal{L}_{\mathrm{stage1}} = \mathcal{L}_{\mathrm{fine}} + \lambda_{\mathrm{b}}(t)\mathcal{L}_{\mathrm{basic}} + \lambda_{\mathrm{h}}(t)\mathcal{L}_{\mathrm{hcons}}. \tag{64}$$

Here, $\mathcal{L}_{\mathrm{fine}}$ is the original fine-grained objective, $\mathcal{L}_{\mathrm{basic}}$ is the cross-entropy loss over basic-level labels, and $\mathcal{L}_{\mathrm{hcons}}$ aligns the family-level distribution induced by the fine-grained logits with the supervised basic-level branch.

In our implementation, hierarchical consistency uses the basic-level branch as a stop-gradient abstraction target:

$$\mathcal{L}_{\mathrm{hcons}} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{KL}\Big(\mathrm{softmax}\left(\mathrm{sg}[u(x_i)]/T\right) \parallel \mathrm{softmax}\left(\mathrm{Agg}(s(x_i))/T\right)\Big). \tag{65}$$

We use temperature $T = 2.0$ in the main CIFAR experiments. Detaching the basic-level logits prevents the consistency loss from destabilizing the family branch and makes the alignment direction consistent with the abstraction-first design.

Algorithm 1 Stage 1 training with basic-level abstraction

Require: training set $\mathcal{D}$, basic taxonomy $(\mathcal{Y}, \mathcal{B}, g)$, model with fine-grained predictor and basic-level branch

for epoch $= 1, \ldots, T_1$ **do**
  update $\lambda_{\mathrm{b}}(t)$ and $\lambda_{\mathrm{h}}(t)$ according to the abstraction-first schedule
  for mini-batch (x, y, b) **do**
    compute fine-grained logits $s(x)$ and basic-level logits $u(x)$
    compute fine-grained loss $\mathcal{L}_{\mathrm{fine}}$
    compute basic-level loss $\mathcal{L}_{\mathrm{basic}} = \mathrm{CE}(u(x), b)$
    compute hierarchical consistency loss $\mathcal{L}_{\mathrm{hcons}}$
    form total loss

$$\mathcal{L}_{\mathrm{stage1}} = \mathcal{L}_{\mathrm{fine}} + \lambda_{\mathrm{b}}(t)\mathcal{L}_{\mathrm{basic}} + \lambda_{\mathrm{h}}(t)\mathcal{L}_{\mathrm{hcons}}$$

    update model parameters by backpropagation
  end for
end for
return best Stage 1 checkpoint according to validation accuracy

Abstraction-first schedule. The weights of the Stage 1 auxiliary losses are scheduled over training. For the main CIFAR-100-LT IR$= 100$ setting, $\lambda_{\mathrm{b}}(t)$ starts large and decreases, while $\lambda_{\mathrm{h}}(t)$ starts at zero and increases later:

$$\lambda_{\mathrm{b}}(t): \quad 0.35 \to 0.35 \to 0.15 \to 0.05, \tag{66}$$

$$\lambda_{\mathrm{h}}(t): \quad 0.00 \to 0.00 \to 0.02 \to 0.05, \tag{67}$$

with milestones at epochs 1, 60, 140, and 200. Linear interpolation is used between consecutive milestones. This schedule first emphasizes explicit family-level learning and then gradually strengthens coarse–fine alignment as the fine-grained predictor becomes more stable.

Algorithm 1 summarizes Stage 1 training.

The best Stage 1 checkpoint is selected by validation accuracy. This checkpoint is then used to initialize the Stage 2 student and to instantiate the frozen Stage 1 teacher.

D.3 Stage 2 Training Details

Stage 2 introduces commitment-guided residual refinement. The best Stage 1 checkpoint is copied twice: one copy is frozen as the teacher, and the other initializes the student. The teacher is used only during Stage 2 training to provide target-aware commitment signals and distillation targets. The student is augmented with the residual refinement branch and is optimized under the Stage 2 objective.

The Stage 2 objective is

$$\begin{aligned}\mathcal{L}_{\mathrm{stage2}} = \mathcal{L}_{\mathrm{fine}} &+ \eta_{\mathrm{b}}(t)\mathcal{L}_{\mathrm{basic}} + \eta_{\mathrm{h}}(t)\mathcal{L}_{\mathrm{hcons}} + \lambda_{\mathrm{s}}(t)\mathcal{L}_{\mathrm{sibling}} \\ &+ \lambda_{\mathrm{df}}\mathcal{L}^{\mathrm{fine}}_{\mathrm{distill}} + \lambda_{\mathrm{db}}\mathcal{L}^{\mathrm{basic}}_{\mathrm{distill}} + \lambda_{\mathrm{r}}\mathcal{L}_{\mathrm{reg}}.\end{aligned} \tag{68}$$

Teacher-based selection. For each training sample (x_i, y_i), the frozen teacher produces fine-grained logits $s^{\mathrm{tea}}(x_i)$ and basic-level logits $u^{\mathrm{tea}}(x_i)$. The target-aware training commitment event is computed according to Eq. (30). A sample receives sibling-level residual supervision only if it satisfies the teacher commitment condition and belongs to a medium-shot or few-shot class:

$$m_i = C^{\mathrm{train}}_{\tau}(x_i) \cdot \mathbb{I}[y_i \in \mathcal{Y}_{\mathrm{mf}}]. \tag{69}$$

In the main CIFAR-100-LT IR$= 100$ setting, we use $\tau = 0.55$ and require coarse agreement between the teacher's basic-level prediction target and the family-level prediction induced by its fine-grained logits.

Sibling-level residual objective. For selected samples, the residual branch is trained using sibling-level cross-entropy over the ground-truth family. The residual logits are centered within the family,

and the target class is mapped to its local index in the sibling set $\mathcal{I}_{b_i}$. Samples with $m_i = 0$ do not contribute to the sibling-level residual loss.

The compact stabilization term $\mathcal{L}_{\text{stab}}$ used in the main text corresponds to

$$\mathcal{L}_{\text{stab}} = \lambda_{\text{df}}\mathcal{L}_{\text{distill}}^{\text{fine}} + \lambda_{\text{db}}\mathcal{L}_{\text{distill}}^{\text{basic}} + \lambda_{\text{r}}\mathcal{L}_{\text{reg}}. \tag{70}$$

Thus, the compact Stage 2 objective in the main text and the expanded objective in this appendix are equivalent.

Distillation. The frozen teacher stabilizes Stage 2 training through fine-grained and basic-level distillation:

$$\mathcal{L}_{\text{distill}}^{\text{fine}} = \frac{1}{N}\sum_{i=1}^{N} \text{KL}\Big(\text{softmax}\left(s^{\text{tea}}(x_i)/T_{\text{d}}\right) \parallel \text{softmax}\left(s^{\text{stu}}(x_i)/T_{\text{d}}\right)\Big), \tag{71}$$

$$\mathcal{L}_{\text{distill}}^{\text{basic}} = \frac{1}{N}\sum_{i=1}^{N} \text{KL}\Big(\text{softmax}\left(u^{\text{tea}}(x_i)/T_{\text{d}}\right) \parallel \text{softmax}\left(u^{\text{stu}}(x_i)/T_{\text{d}}\right)\Big). \tag{72}$$

We use $T_{\text{d}} = 2.0$, $\lambda_{\text{df}} = 0.5$, and $\lambda_{\text{db}} = 0.2$ in the main CIFAR experiments.

Residual regularization. Let $W \in \mathbb{R}^{K\times d}$ be the residual-classifier weight matrix and W_y its row for fine class y. With training count n_y, $n_{\max} = \max_y n_y$, and $\epsilon_n = 10^{-6}$, the normalized frequency weight is

$$a_y = \frac{\left(n_{\max}/(n_y + \epsilon_n)\right)^{1/2}}{\frac{1}{K}\sum_{c=1}^{K}\left(n_{\max}/(n_c + \epsilon_n)\right)^{1/2}}. \tag{73}$$

The two penalties are

$$\mathcal{L}_{\text{residual-}\ell_2} = \frac{1}{K}\sum_{y=1}^{K} a_y \frac{\|W_y\|_2^2}{d}, \tag{74}$$

$$\mathcal{L}_{\text{residual-family}} = \frac{1}{M}\sum_{b=1}^{M}\frac{1}{d}\left\|\frac{1}{|\mathcal{I}_b|}\sum_{y\in\mathcal{I}_b} W_y\right\|_2^2. \tag{75}$$

Their complete contribution to the optimization objective is

$$\mathcal{R}_{\text{res}} = \lambda_{\ell_2}\mathcal{L}_{\text{residual-}\ell_2} + \lambda_{\text{fam}}\mathcal{L}_{\text{residual-family}}, \qquad \lambda_{\ell_2} = \lambda_{\text{fam}} = 5\times 10^{-4}. \tag{76}$$

The compact product $\lambda_{\text{r}}\mathcal{L}_{\text{reg}}$ in the main objective denotes $\mathcal{R}_{\text{res}}$; the two displayed coefficients are effective weights and are not multiplied by a second regularization coefficient. Both penalties act on classifier parameters, not on per-sample residual logits or output probabilities. They are evaluated once per optimization step over all K residual-classifier rows, without the sample-selection mask and without α_{r}. The family term penalizes the mean residual weight direction within a family; it is not a KL penalty or a guarantee that family probability mass is unchanged.

The sample mask controls which examples supply sibling supervision. It does not remove head-class rows from a selected sibling set: a head class can still appear as a competing logit when a selected lower-frequency example belongs to the same family. This distinction also matters for interpreting many-shot performance.

Stage 2 schedule. In Stage 2, the abstraction-related losses are retained with reduced weights. For the main CIFAR-100-LT IR$=100$ setting, the schedules are

$$\eta_{\text{b}}(t): \quad 0.12 \to 0.08 \to 0.04, \tag{77}$$

$$\eta_{\text{h}}(t): \quad 0.02 \to 0.03 \to 0.04, \tag{78}$$

$$\lambda_{\text{s}}(t): \quad 0.02 \to 0.03 \to 0.05 \to 0.06, \tag{79}$$

with milestones at epochs 1, 20, and 80 for η_{b} and η_{h}, and epochs 1, 10, 40, and 80 for λ_{s}. Linear interpolation is used between milestones.

Algorithm 2 Stage 2 training with teacher-based selective residual refinement

Require: best Stage 1 checkpoint, training set $\mathcal{D}$, basic taxonomy $(\mathcal{Y}, \mathcal{B}, g)$
initialize frozen teacher from the best Stage 1 checkpoint
initialize student from the same Stage 1 checkpoint
add residual refinement branch to the student
for epoch $= 1, \ldots, T_2$ **do**
    update $\eta_{\mathrm{b}}(t)$, $\eta_{\mathrm{h}}(t)$, and $\lambda_{\mathrm{s}}(t)$
    if epoch is in the residual warmup period **then**
        freeze the main prediction pathway and train the residual branch
    else
        optimize the full student
    end if
    for mini-batch (x, y, b) **do**
        teacher computes $s^{\mathrm{tea}}(x)$ and $u^{\mathrm{tea}}(x)$
        compute teacher-based training mask m^{train}
        student computes $s^{\mathrm{stu}}(x)$, $u^{\mathrm{stu}}(x)$, and residual logits $r(x)$
        compute fine-grained loss $\mathcal{L}_{\mathrm{fine}}$
        compute retained abstraction losses $\mathcal{L}_{\mathrm{basic}}$ and $\mathcal{L}_{\mathrm{hcons}}$
        compute sibling residual loss $\mathcal{L}_{\mathrm{sibling}}$ on selected samples
        compute teacher-student distillation losses
        compute residual regularization
        form total loss $\mathcal{L}_{\mathrm{stage2}}$
        update student parameters by backpropagation
    end for
end for
return best Stage 2 student checkpoint according to validation accuracy

At the beginning of Stage 2, we use a 10-epoch warmup period. During this period, the main prediction pathway is kept fixed and the residual branch is trained to stabilize. After warmup, the full student is optimized jointly. This prevents the newly introduced residual branch from destabilizing the fine-grained predictor at the start of Stage 2.

Algorithm 2 summarizes Stage 2 training.

D.4 Commitment Thresholds and Selection Rules

The commitment mechanism controls when residual refinement is trained and activated. We use separate thresholds for training and evaluation because the available information differs in the two phases.

Training-time selection. During Stage 2 training, labels are available. We therefore use the target-aware teacher commitment event defined in Eq. (30). A sample is selected for residual supervision if the teacher assigns sufficiently high probability to the ground-truth family and if the family-level prediction induced by the teacher's fine-grained logits agrees with the ground-truth family. The selection is further restricted to medium-shot and few-shot classes.

For the main CIFAR-100-LT IR$= 100$ setting, the training threshold is

$$\tau = 0.55. \tag{80}$$

Evaluation-time selection. At test time, labels are unavailable. We therefore use the prediction-based student commitment event defined in Eq. (33). The residual correction is activated only when the student basic-level branch is sufficiently confident and agrees with the family-level prediction induced by the fine-grained logits.

For the main CIFAR-100-LT IR$= 100$ setting, the evaluation threshold is

$$\tau_{\mathrm{eval}} = 0.60. \tag{81}$$

In addition, residual correction is applied only when the base fine-grained prediction belongs to the medium-shot or few-shot class set:

$$\hat{y}_{\mathrm{base}}(x) \in \mathcal{Y}_{\mathrm{mf}}. \tag{82}$$

This mirrors the Stage 2 training objective, which focuses residual supervision on low-frequency classes.

Development protocol. The operating point is chosen once during preliminary development using development-set Top-1 accuracy. Lower residual activation is used only to break near-ties. The candidate grids are

$$\tau \in \{0.45, 0.50, 0.55, 0.60, 0.65\}, \tag{83}$$

$$\tau_{\mathrm{eval}} \in \{0.40, 0.50, 0.60, 0.70, 0.80\}, \tag{84}$$

$$\alpha_{\mathrm{r}} \in \{0, 0.02, 0.04, 0.06, 0.08\}. \tag{85}$$

The selected constants are fixed across the reported seeds, not learned by the model or separately tuned for each run. CIFAR-100-LT and the large-scale datasets use $\tau/\tau_{\mathrm{eval}} = 0.55/0.60$; CIFAR-10-LT uses 0.65/0.70. Target-aware frozen-teacher selection allows a less restrictive training threshold, while label-free student activation is more conservative. Threshold selection is distinct from taxonomy construction: the mappings are fixed before training and are not chosen by recognition performance.

Threshold sensitivity. The threshold controls a coverage–reliability trade-off. Lower thresholds activate residual correction on more samples but may include unreliable family predictions. Higher thresholds improve reliability but may miss useful correction opportunities. We therefore evaluate threshold sensitivity in Appendix F.12. The main threshold values above are fixed for the reported main results.

D.5 Inference Procedure

At inference time, only the trained Stage 2 student is used. The frozen teacher is not used. The student first produces fine-grained logits before residual merging, denoted by

$$s_{\mathrm{base}}(x), \tag{86}$$

basic-level logits

$$u^{\mathrm{stu}}(x), \tag{87}$$

and residual logits

$$r(x). \tag{88}$$

The student then computes the evaluation commitment event $C^{\mathrm{eval}}_{\tau_{\mathrm{eval}}}(x)$ and the evaluation mask $m^{\mathrm{eval}}(x)$. If the mask is zero, the model outputs the base logits:

$$s^{\mathrm{final}}(x) = s_{\mathrm{base}}(x). \tag{89}$$

If the mask is one, the residual is centered within the committed family $\hat{b}_u(x)$ and merged into the base logits:

$$s^{\mathrm{final}}(x) = s_{\mathrm{base}}(x) + \alpha_{\mathrm{r}}\tilde{r}(x; \hat{b}_u(x)). \tag{90}$$

For the main CIFAR-100-LT IR$= 100$ setting, the residual merge coefficient is scheduled during Stage 2 validation and reaches

$$\alpha_{\mathrm{r}} = 0.06 \tag{91}$$

at the end of training. The same value is used for the final evaluation of the selected Stage 2 checkpoint.

Algorithm 3 summarizes inference with selective residual correction.

This inference rule makes the residual branch conservative. The model falls back to the base fine-grained prediction whenever family-level commitment is unreliable or when the base prediction does not belong to the target low-frequency set.

D.6 Reproducibility Details

We include the following artifacts for reproducibility.

Algorithm 3 Inference with commitment-guided residual correction

Require: test sample x, trained Stage 2 student, threshold τ_{eval}, residual strength α_{r}
compute base fine-grained logits $s_{\mathrm{base}}(x)$
compute basic-level logits $u^{\mathrm{stu}}(x)$
compute residual logits $r(x)$
compute $\hat{b}_u(x) = \arg\max_b u_b^{\mathrm{stu}}(x)$
compute $\hat{b}_s(x) = \arg\max_b \mathrm{Agg}(s_{\mathrm{base}}(x))_b$
compute evaluation commitment $C_{\tau_{\mathrm{eval}}}^{\mathrm{eval}}(x)$
compute base prediction $\hat{y}_{\mathrm{base}}(x) = \arg\max_y s_{\mathrm{base},y}(x)$
if $C_{\tau_{\mathrm{eval}}}^{\mathrm{eval}}(x) = 1$ and $\hat{y}_{\mathrm{base}}(x) \in \mathcal{Y}_{\mathrm{mf}}$ **then**
    compute centered residual $\tilde{r}(x; \hat{b}_u(x))$
    output $s^{\mathrm{final}}(x) = s_{\mathrm{base}}(x) + \alpha_{\mathrm{r}} \tilde{r}(x; \hat{b}_u(x))$
else
    output $s^{\mathrm{final}}(x) = s_{\mathrm{base}}(x)$
end if

Configuration files. Each benchmark has separate configuration files for Stage 1 and Stage 2. The configuration records the model architecture, optimizer, learning-rate schedule, batch size, augmentation policy, loss weights, commitment thresholds, residual regularization weights, and checkpoint paths.

Taxonomy files. Each benchmark has a fixed fine-to-basic mapping file. For CIFAR-100-LT, the mapping is the official superclass annotation. For CIFAR-10-LT and Places-LT, the mapping files contain the fixed semantic groupings used in the experiments. For ImageNet-LT and iNaturalist 2018, the mapping files contain the hierarchy-derived family assignments described in Appendix C.

Checkpoint selection. Stage 1 uses the checkpoint with the best validation accuracy as both the initialization for the Stage 2 student and the frozen Stage 2 teacher. Stage 2 also selects the best checkpoint by validation accuracy. The final test results are reported using the selected Stage 2 student.

Random seeds. Unless otherwise specified, all main-table BASICLT results and the core ablations in Tables 3–4 are reported as mean $\pm$ standard deviation over five independent runs. The seed controls model initialization, data shuffling, and stochastic augmentation. Additional appendix sensitivity analyses that are reported without uncertainty ranges are run under the fixed seed specified in the corresponding configuration.

Evaluation. All final evaluations are conducted on the original fine-grained label space. The basic-level branch and residual branch are used only to support abstraction and selective refinement. Reported many-shot, medium-shot, and few-shot accuracies are computed according to the standard class-frequency partitions of each benchmark.

Implementation distinction. The implementation separates the original fine-grained prediction pathway from the components introduced by BASICLT. The basic-level branch, hierarchical consistency loss, teacher-based commitment selection, residual refinement branch, and selective residual inference rule are the components introduced by the proposed framework. Controlled ablations enable or disable these components while keeping the remaining training and evaluation protocol unchanged.

Computational resources. We also record the hardware and wall-clock cost used for the reported experiments. Unless otherwise specified, the controlled CIFAR experiments are conducted on a single NVIDIA RTX 4090 GPU. Under the main CIFAR-100-LT IR$= 100$ configuration in Table 12, Stage 1 training takes approximately 1.0–1.3 GPU-hours and Stage 2 training takes approximately 0.5–0.7 GPU-hours, resulting in about 1.5–2.0 GPU-hours for a full two-stage run. The peak GPU memory usage is approximately 1.3 GB for Stage 1 and 0.6 GB for Stage 2. The additional training cost of BASICLT mainly comes from the second-stage residual refinement, while the basic-level branch and

Table 15: Separation between the underlying fine-grained prediction pathway and components introduced by BASICLT. Controlled ablations keep the underlying pathway fixed and vary only the proposed components.

Part	Role
Fine-grained predictor	Original recognition pathway over $\mathcal{Y}$
Original fine-grained objective	Trains the base prediction pathway
Basic-level branch	Introduced by BASICLT for family-level abstraction
Hierarchical consistency	Introduced by BASICLT for coarse–fine alignment
Frozen Stage 1 teacher	Introduced by BASICLT for Stage 2 commitment and distillation
Family-conditioned residual branch	Introduced by BASICLT for sibling-level correction
Commitment-guided selection	Introduced by BASICLT for selective residual training and inference

residual branch introduce only a small overhead relative to the underlying fine-grained predictor. At inference time, the extra computation consists of one basic-level prediction, a commitment check, and a conditional residual merge; when the commitment criterion is not satisfied, the model falls back to the base fine-grained prediction.

Overall, these details are intended to make the reported results auditable. The key reproducibility requirements are the fixed taxonomy mapping, the two-stage checkpoint protocol, the teacher-based training mask, the residual centering rule, and the evaluation-time commitment rule.

D.7 Loss Roles and Schedule Sensitivity

The fine-grained loss preserves the target task, basic supervision learns the family scaffold, and hierarchical consistency aligns the two predictions. In Stage 2, sibling cross-entropy trains additional differentiation, distillation preserves the teacher's prediction structure, and the parameter penalties control residual weights. The per-loss schedules are fixed configuration choices, not independently learned control policies.

The available controls distinguish these roles from secondary stabilization. Removing the Stage 1 curriculum changes overall accuracy from 56.1 ± 0.3 to 55.3 ± 0.4 in Table 19. With teacher initialization, removing Stage 2 warmup changes 60.59 ± 0.36 to 60.52 ± 0.38 in Table 28. In contrast, random refinement initialization with warmup gives 47.17 ± 0.62. These controls support the importance of starting from the learned Stage 1 model, while showing that warmup is a comparatively small stabilizer in that experiment. They do not establish that every schedule, weight, or joint perturbation is interchangeable. The reported recipe retains its stated schedules.

Learning-rate warmup and the residual-only main-path freeze are separate operations. During the latter, the base pathway is fixed while residual parameters are trained. Afterward, the student follows the joint objective. At inference, the teacher is absent and the residual mask governs merging; the mask alone does not establish that the residual forward computation has been skipped.

E Extended Results and Evaluation Scope

This appendix clarifies the scope of the extended evaluation and summarizes additional observations that complement the main results. The main paper reports two types of quantitative comparisons. First, Table 1 provides detailed many-shot, medium-shot, few-shot, and overall results on CIFAR-100-LT under multiple imbalance ratios. Second, Table 2 provides a cross-benchmark comparison on CIFAR-10-LT, ImageNet-LT, iNaturalist 2018, and Places-LT.

We do not construct a large unified shot-wise comparison table for all methods on iNaturalist 2018 and Places-LT in the appendix. Many published methods report only overall accuracy on these datasets, and do not provide consistent many-shot, medium-shot, and few-shot breakdowns under the same evaluation protocol. Filling such a table with incomplete entries would make the comparison difficult to interpret and could overstate the availability of directly comparable results. Instead, we use the appendix for two purposes: to clarify the evaluation scope and to provide controlled ablations and analyses under matched settings.

E.1 Evaluation Scope

The benchmarks used in this work differ in how completely prior results are reported.

CIFAR-100-LT. CIFAR-100-LT is the primary controlled benchmark for detailed analysis. It has an official superclass taxonomy, a small and balanced family structure, and a widely used long-tailed construction protocol. For this reason, the main paper reports full results across imbalance ratios 100, 50, and 10, together with many-shot, medium-shot, few-shot, and overall accuracy.

CIFAR-10-LT. CIFAR-10-LT provides a smaller controlled setting. Although its taxonomy is shallow, it is useful for testing whether the proposed abstraction-to-differentiation mechanism remains effective when only a coarse semantic partition is available. The main paper reports many-shot, medium-shot, few-shot, and overall accuracy for CIFAR-10-LT under IR$= 100$.

ImageNet-LT. ImageNet-LT provides a large-scale object recognition setting with a heterogeneous label space. Since many prior methods report shot-wise breakdowns on ImageNet-LT, the main paper includes many-shot, medium-shot, few-shot, and overall comparisons. This benchmark is important for testing whether the method scales beyond controlled CIFAR settings and whether a WordNet-derived basic taxonomy can support abstraction and selective differentiation.

iNaturalist 2018. iNaturalist 2018 is a large-scale fine-grained benchmark with a natural biological hierarchy. Many published long-tailed recognition methods report only overall top-1 accuracy on this dataset. Because full shot-wise breakdowns are not consistently available across prior methods, we report overall comparisons in the main paper and use controlled analyses to study the behavior of BASICLT.

Places-LT. Places-LT evaluates scene-centric long-tailed recognition. As with iNaturalist 2018, many prior methods report only overall top-1 accuracy, and complete shot-wise breakdowns are not always available. We therefore report overall comparison in the main paper and avoid constructing an incomplete shot-wise SOTA table in the appendix.

E.2 Summary of Main Empirical Findings

The main results support three observations.

First, BASICLT performs consistently across benchmark types. The evaluated datasets include controlled object recognition benchmarks, a large-scale object benchmark, a fine-grained biological benchmark, and a scene-centric benchmark. The improvement across these settings suggests that the proposed abstraction-to-differentiation framework is not tied to a single dataset type.

Second, the gains are most consistent in lower-frequency regimes. On benchmarks where many-shot, medium-shot, and few-shot breakdowns are available, BASICLT shows particularly consistent improvements in medium-shot and few-shot categories. This pattern is aligned with the motivation that scarce classes can benefit from first learning stable family-level abstraction before additional sibling-level differentiation is introduced.

Third, overall improvements are supported by controlled ablations. The ablation studies in Appendix F show that the gains cannot be explained only by additional training time, extra parameters, or an auxiliary prediction head. Instead, the strongest results are obtained when the added capacity is organized by a meaningful basic taxonomy and activated through commitment-guided selective refinement.

E.3 Why Extended Ablations Are More Informative Than Incomplete SOTA Tables

For iNaturalist 2018 and Places-LT, incomplete reporting in prior work makes a fully detailed cross-method shot-wise table less informative. A table containing many missing entries can obscure the central question of this paper: whether semantic granularity control improves long-tailed recognition beyond conventional fine-grained training.

We therefore focus the appendix on controlled comparisons under matched conditions. These ablations directly test the components of BASICLT:

- whether basic-level abstraction improves the fine-grained predictor;
- whether hierarchical consistency is necessary;
- whether residual refinement should be selective rather than uniform;
- whether teacher-based commitment is more stable than student-side selection;
- whether the gain depends on meaningful taxonomy construction;
- whether the improvement remains after controlling for training budget and parameter count.

These controlled experiments are more directly tied to the claims of the paper than an incomplete table of externally reported large-scale results.

F Extended Ablation Studies

This appendix provides additional ablation studies for BASICLT. The main paper reports the core component ablation and the refinement policy ablation. Here, we expand the analysis to cover training budget, parameter count, Stage 1 objectives, Stage 2 selection rules, residual design, distillation, threshold sensitivity, target group selection, taxonomy construction, and cross-benchmark selective refinement.

Unless otherwise specified, ablations are conducted on CIFAR-100-LT with IR$=100$. This setting is used because it has an official basic taxonomy and provides a controlled environment for studying long-tailed scarcity. All variants use the same dataset, fine-to-basic mapping, data augmentation, evaluation protocol, and underlying fine-grained predictor unless explicitly stated otherwise.

F.1 Overview of Ablation Groups

The ablation studies are organized into the following groups.

- **Core component ablations** examine the effect of the basic branch, hierarchical consistency, Stage 2 refinement, and teacher-student stabilization.
- **Refinement policy ablations** compare uniform refinement, confidence-only selection, agreement-only selection, and the full commitment-guided policy.
- **Training budget and parameter controls** test whether the improvement is caused merely by longer training or additional parameters.
- **Stage 1 objective ablations** study the roles of basic-level supervision, hierarchical consistency, stop-gradient alignment, and the abstraction-first schedule.
- **Stage 2 selection ablations** compare teacher-based selection, student-side selection, target group restrictions, and evaluation-time eligibility rules.
- **Residual design ablations** study residual centering, within-family masking, residual regularization, and residual strength.
- **Distillation and warmup ablations** analyze the importance of fine-grained distillation, basic-level distillation, and residual warmup.
- **Taxonomy sensitivity** evaluates semantic, coarse, fine, random, shuffled, and noisy taxonomies.
- **Cross-benchmark refinement analysis** checks whether selective refinement remains preferable to uniform refinement beyond CIFAR-100-LT.

F.2 Core Component Ablation

Table 16 repeats the core component ablation from the main paper for completeness. The base predictor is trained without the proposed basic-level abstraction and residual refinement. Adding the basic branch improves all shot groups, indicating that explicit family-level supervision is useful even before Stage 2. The full Stage 1 model further improves performance by adding hierarchical consistency. Stage 2 without distillation improves medium-shot and few-shot performance but hurts

Table 16: Core component ablation on **CIFAR-100-LT** (IR$=100$). Results are reported as mean $\pm$ standard deviation over five runs.

Variant	Many	Medium	Few	All
Base predictor	69.5 ± 0.4	52.4 ± 0.5	30.2 ± 0.6	52.6 ± 0.2
+ Basic branch	70.3 ± 0.5	54.1 ± 0.3	31.8 ± 0.7	54.2 ± 0.5
+ Stage 1 full	72.1 ± 0.3	56.4 ± 0.6	33.5 ± 0.6	56.1 ± 0.4
+ Stage 2 w/o distill.	66.8 ± 0.4	58.2 ± 0.2	35.1 ± 0.3	55.4 ± 0.1
+ Stage 2 w/ distill.	75.3 ± 0.2	60.8 ± 0.4	35.8 ± 0.5	58.7 ± 0.5
Full BASICLT	$\mathbf{78.4\pm0.3}$	$\mathbf{63.2\pm0.4}$	$\mathbf{37.0\pm0.5}$	$\mathbf{60.6\pm0.4}$

Table 17: Refinement policy ablation on CIFAR-100-LT (IR$=100$).

Policy	Many	Medium	Few	All
Stage 1 only	72.1 ± 0.4	56.4 ± 0.5	33.5 ± 0.2	56.1 ± 0.5
Uniform refinement	70.4 ± 0.6	57.0 ± 0.5	34.2 ± 0.7	55.8 ± 0.5
Confidence-only	74.8 ± 0.4	59.3 ± 0.4	35.2 ± 0.6	58.2 ± 0.4
Agreement-only	75.5 ± 0.4	60.2 ± 0.4	35.9 ± 0.6	59.0 ± 0.4
Commitment-guided	$\mathbf{78.4\pm0.3}$	$\mathbf{63.2\pm0.4}$	$\mathbf{37.0\pm0.5}$	$\mathbf{60.6\pm0.4}$

many-shot accuracy, showing that unconstrained refinement can interfere with the base predictor. Adding teacher-student stabilization improves the trade-off, and the full model obtains the best overall result.

These results indicate that the final gain is not produced by a single component. Stage 1 provides a better abstraction-aware representation, while Stage 2 provides additional within-family differentiation when guided by reliable commitment.

F.3 Refinement Policy Ablation

Table 17 compares different refinement policies. Uniform refinement applies the residual branch without commitment-based selection. Confidence-only refinement uses only the basic-level confidence threshold. Agreement-only refinement uses only agreement between the basic-level branch and the family-level prediction induced by the fine-grained logits. The full commitment-guided policy uses both signals.

Uniform refinement is less effective because it applies sibling-level correction even when family-level structure is unreliable. Confidence and agreement each improve the selection quality, but the best result is obtained when both are used. This supports the commitment design used by BASICLT.

F.4 Training Budget and Parameter Controls

A potential concern is that BASICLT may improve simply because it uses two-stage training or introduces additional parameters. We therefore include controls that match the training budget and parameter count.

The same-budget base predictor improves only marginally over the standard base predictor, indicating that longer training alone does not explain the gain. The parameter-matched auxiliary head also falls short of BASICLT, suggesting that the improvement is not merely caused by additional parameters. The strongest result requires the combination of basic-level abstraction and commitment-guided residual refinement.

F.5 Stage 1 Objective Ablation

We study the contributions of the Stage 1 abstraction components in Table 19. The basic-level supervision term teaches the model explicit family-level structure, while the hierarchical consistency term aligns the family-level distribution induced by the fine-grained predictor with the basic-level branch. We also evaluate the effect of stop-gradient alignment and the abstraction-first schedule.

Table 18: Training budget and parameter controls on CIFAR-100-LT (IR$=100$). "Same-budget" trains the base predictor for the same total number of epochs as the two-stage pipeline. "Param-matched auxiliary head" adds comparable parameters without the proposed taxonomy-conditioned residual design.

Variant	Stage 1	Stage 2	Extra params	Selective residual	All
Base predictor	200	0	No	No	52.6
Base predictor, same-budget	280	0	No	No	53.1
Base + param-matched auxiliary head	200	80	Yes	No	54.0
Stage 1 full	200	0	Basic branch	No	56.1
Stage 1 + uniform residual	200	80	Yes	No	55.8
Full BasicLT	200	80	Yes	Yes	**60.6**

Table 19: Stage 1 objective ablation on CIFAR-100-LT (IR$=100$). Results are reported as mean $\pm$ standard deviation over repeated runs.

Variant	Many	Medium	Few	All
Base predictor	69.5 ± 0.4	52.4 ± 0.5	30.2 ± 0.6	52.6 ± 0.4
+ Basic supervision only	70.3 ± 0.4	54.1 ± 0.4	31.8 ± 0.6	54.2 ± 0.3
+ Hierarchical consistency only	70.0 ± 0.4	53.6 ± 0.5	31.1 ± 0.6	53.8 ± 0.4
Stage 1 w/o stop-gradient	70.8 ± 0.5	55.0 ± 0.5	32.6 ± 0.6	55.0 ± 0.4
Stage 1 w/o curriculum	71.2 ± 0.4	55.2 ± 0.5	32.9 ± 0.6	55.3 ± 0.4
Stage 1 full	$\mathbf{72.1\pm0.3}$	$\mathbf{56.4\pm0.4}$	$\mathbf{33.5\pm0.5}$	$\mathbf{56.1\pm0.3}$

Both basic-level supervision and hierarchical consistency are beneficial. Removing stop-gradient weakens the result, suggesting that treating the supervised basic branch as a stable abstraction target is useful. Removing the abstraction-first schedule also reduces performance, indicating that it is helpful to emphasize basic-level abstraction early and strengthen consistency later.

F.6 Hierarchical Consistency Design

We further study the design of hierarchical consistency. In the main model, we align the fine-grained predictor's induced family distribution to the stop-gradient basic-level branch:

$$\mathrm{KL}(\mathrm{softmax}(\mathrm{sg}[u]/T) \parallel \mathrm{softmax}(\mathrm{Agg}(s)/T)) . \tag{92}$$

Table 20 compares this design with alternative directions.

The stop-gradient basic-to-fine direction performs best. This is consistent with the design of Stage 1: the basic-level branch is directly supervised by family labels and can provide a stable abstraction target, while the fine-grained predictor is encouraged to become consistent with this coarse semantic structure.

F.7 Teacher-Based versus Student-Side Selection

The main Stage 2 training procedure uses the frozen Stage 1 teacher to compute the training-time commitment mask. We compare this with a student-side selection variant, where the evolving student computes the mask during training.

Teacher-based selection is more stable because the teacher does not change during Stage 2. Student-side selection allows the evolving student base and basic predictions to change the selection set; the frozen teacher removes this source of feedback. The full teacher commitment rule performs best because it combines target-family confidence with coarse agreement.

F.8 Target Group Selection

Stage 2 residual refinement is primarily designed for medium-shot and few-shot classes. Table 22 evaluates different target group choices.

Table 22 shows that different target-group choices lead to close overall performance, but with different activation patterns. Refining all classes activates the residual branch much more frequently and

Table 20: Hierarchical consistency design on **CIFAR-100-LT** (IR$=100$). Results are reported as mean $\pm$ standard deviation over repeated runs.

Variant	Many	Medium	Few	All
No hierarchical consistency	70.3 ± 0.4	54.1 ± 0.4	31.8 ± 0.6	54.2 ± 0.3
Symmetric KL	71.0 ± 0.4	55.3 ± 0.5	32.6 ± 0.6	55.2 ± 0.4
Fine-to-basic direction	70.6 ± 0.4	54.8 ± 0.5	32.1 ± 0.6	54.8 ± 0.4
Basic-to-fine, no stop-gradient	70.8 ± 0.5	55.0 ± 0.5	32.6 ± 0.6	55.0 ± 0.4
Basic-to-fine, stop-gradient	$\mathbf{72.1\pm0.3}$	$\mathbf{56.4\pm0.4}$	$\mathbf{33.5\pm0.5}$	$\mathbf{56.1\pm0.3}$

Table 21: Teacher-based versus student-side selection in Stage 2 on CIFAR-100-LT (IR$=100$). Results are reported as mean $\pm$ standard deviation over repeated runs.

Stage 2 selection source	Many	Medium	Few	All
Student-side selection	75.9 ± 0.5	61.1 ± 0.5	36.0 ± 0.7	59.0 ± 0.5
Teacher confidence only	76.7 ± 0.4	61.9 ± 0.4	36.3 ± 0.6	59.5 ± 0.4
Teacher agreement only	77.2 ± 0.4	62.1 ± 0.4	36.5 ± 0.6	59.8 ± 0.4
Teacher commitment	$\mathbf{78.4\pm0.3}$	$\mathbf{63.2\pm0.4}$	$\mathbf{37.0\pm0.5}$	$\mathbf{60.6\pm0.4}$

improves many-shot accuracy, but it reduces medium-shot accuracy and does not improve the overall result. Restricting refinement to low-frequency groups is therefore more conservative and better aligned with the purpose of Stage 2. In this run, the few-only variant obtains the best overall accuracy, while the medium+few variant obtains the best medium-shot accuracy and remains a balanced default. We use medium+few as the main configuration because it covers both lower-frequency regimes while avoiding broad residual activation on many-shot predictions.

The few-only setting has the highest overall mean in this comparison, while medium+few has the highest medium-shot mean. All-class refinement improves many-shot accuracy but substantially increases activation without increasing the overall mean. Medium+few is therefore the chosen coverage–activation compromise, not a claim of strict numerical optimality for every metric.

F.9 Evaluation-Time Eligibility Rule

At inference time, true labels are unavailable. The main model activates residual correction only when the student commitment criterion is satisfied and the base prediction belongs to the target low-frequency set. We evaluate alternatives in Table 23.

The commitment-only variant activates refinement more broadly and can affect many-shot predictions. The predicted low-frequency rule alone is not sufficient because it does not verify family-level reliability. Combining both criteria yields the best performance.

F.10 Residual Design Ablation

We evaluate the design of the residual refinement branch in Table 24. The main model uses a family-centered and family-masked residual correction. We compare it with global residual correction, uncentered residual correction, and variants without residual regularization.

Global residual correction is less effective because it can alter scores across unrelated families. Family masking improves performance by restricting correction to the selected family. Centering further encourages the residual branch to model relative sibling-level differences rather than global class bias. Residual regularization also contributes by preventing the correction from becoming overly aggressive.

F.11 Residual Strength Sensitivity

The residual merge coefficient α_{r} controls the strength of the residual correction at inference. Table 25 reports sensitivity to this coefficient.

Table 22: Target group selection for Stage 2 residual refinement on **CIFAR-100-LT** (IR$=100$). The selection ratio denotes the fraction of evaluation samples on which residual refinement is activated. Results are reported as mean $\pm$ standard deviation over repeated runs.

Target groups	Selection ratio (%)	Many	Medium	Few	All
Few only	2.19 ± 0.18	78.91 ± 0.35	62.74 ± 0.42	$\mathbf{37.27\pm0.55}$	$\mathbf{60.76\pm0.38}$
Medium + few	14.70 ± 0.82	78.46 ± 0.32	$\mathbf{63.11\pm0.40}$	36.80 ± 0.50	60.59 ± 0.36
All classes	39.17 ± 1.24	$\mathbf{79.60\pm0.41}$	61.40 ± 0.48	$\mathbf{37.27\pm0.58}$	60.53 ± 0.42

Table 23: Evaluation-time eligibility rule on CIFAR-100-LT (IR$=100$). Results are reported as mean $\pm$ standard deviation over repeated runs.

Inference eligibility	Many	Medium	Few	All
No residual at inference	75.3 ± 0.4	60.8 ± 0.4	35.8 ± 0.6	58.7 ± 0.4
Commitment only	76.9 ± 0.4	62.8 ± 0.4	36.8 ± 0.6	60.0 ± 0.4
Predicted low-frequency only	75.6 ± 0.5	61.4 ± 0.5	36.1 ± 0.6	59.1 ± 0.5
Commitment + predicted low-frequency	$\mathbf{78.4\pm0.3}$	$\mathbf{63.2\pm0.4}$	$\mathbf{37.0\pm0.5}$	$\mathbf{60.6\pm0.4}$

A small positive residual strength improves performance, while an overly large coefficient can make the correction too aggressive. We use $\alpha_{\mathrm{r}} = 0.06$ in the main CIFAR-100-LT IR$=100$ setting.

F.12 Commitment Threshold Sensitivity

The evaluation commitment threshold τ_{eval} controls the coverage–reliability trade-off. Table 26 reports performance under different thresholds.

Lower thresholds activate refinement more often but can include unreliable samples. Higher thresholds are more conservative but miss useful refinement opportunities. The intermediate threshold performs best.

F.13 Distillation Variants

Table 27 studies the effect of teacher-student distillation in Stage 2. Table 27 shows that distillation is important for stabilizing Stage 2 residual refinement. Removing distillation reduces overall accuracy by about one percentage point. Fine-level distillation provides most of the stabilization benefit and obtains the best few-shot and overall accuracy in this run. Adding basic-level distillation slightly improves many-shot and medium-shot accuracy, but does not improve few-shot accuracy here. This suggests that fine-grained distillation is the dominant stabilizing factor, while basic-level distillation mainly helps preserve the family-level structure learned in Stage 1.

F.14 Teacher Initialization and Warmup Design

Stage 2 initializes the student from the best Stage 1 checkpoint and freezes the same checkpoint as the teacher. We also use a short warmup period during which the residual branch is trained while the main prediction pathway is kept fixed.

Table 28 shows that initialization from the Stage 1 teacher is critical. Without teacher initialization, overall accuracy drops by more than ten percentage points, and few-shot accuracy drops by more than twelve percentage points. This confirms that Stage 2 should refine a learned abstraction rather than train the residual pathway from an unstable initialization. Warmup is helpful but secondary in this run: with teacher initialization, removing warmup changes overall accuracy by only 0.07 percentage points. We keep the warmup in the default configuration because it is conservative and stabilizes the newly introduced residual branch, even though its quantitative effect is much smaller than that of teacher initialization.

Table 24: Residual design ablation on **CIFAR-100-LT** (IR$=100$). Results are reported as mean $\pm$ standard deviation over repeated runs.

Residual design	Many	Medium	Few	All
Global residual correction	72.8 ± 0.6	58.4 ± 0.5	34.6 ± 0.7	56.8 ± 0.5
Family-masked, uncentered	75.0 ± 0.5	60.5 ± 0.5	35.7 ± 0.6	58.6 ± 0.4
Centered residual, no regularization	76.2 ± 0.5	61.4 ± 0.5	36.2 ± 0.6	59.2 ± 0.4
Centered residual, no family mask	74.1 ± 0.6	60.2 ± 0.5	35.5 ± 0.7	58.0 ± 0.5
Family-masked centered residual	$\mathbf{78.4\pm0.3}$	$\mathbf{63.2\pm0.4}$	$\mathbf{37.0\pm0.5}$	$\mathbf{60.6\pm0.4}$

Table 25: Sensitivity to the residual merge coefficient α_{r} on CIFAR-100-LT (IR$=100$). Results are reported as mean $\pm$ standard deviation over repeated runs.

α_{r}	0.00	0.02	0.04	0.06	0.08
Overall accuracy	58.7 ± 0.4	59.4 ± 0.4	60.1 ± 0.4	$\mathbf{60.6\pm0.4}$	60.0 ± 0.5
Few-shot accuracy	35.8 ± 0.6	36.3 ± 0.6	36.8 ± 0.5	$\mathbf{37.0\pm0.5}$	36.6 ± 0.6

F.15 Uniform versus Selective Refinement across Benchmarks

Table 29 compares Stage 1, uniform refinement, and selective refinement across multiple benchmarks. For datasets where full shot-wise breakdowns are unavailable for all prior methods, this table reports controlled variants of our own method under the same evaluation protocol.

Selective refinement consistently outperforms uniform refinement. This supports the claim that the residual branch should be activated under reliable family-level commitment rather than applied to all samples.

F.16 Taxonomy Sensitivity

Table 30 studies the effect of taxonomy construction on CIFAR-100-LT. The official superclass taxonomy provides a semantically meaningful and balanced reference. We compare it with coarser, finer, random, shuffled, and noisy taxonomies.

The official semantic taxonomy performs best. Random and shuffled groupings are less effective, showing that the gain does not come merely from adding an auxiliary branch with fewer labels. Coarser and finer taxonomies also underperform the official taxonomy, supporting the importance of moderate semantic granularity.

F.17 Family Error and Sibling Error Decomposition

To better understand the mechanism of BASICLT, we decompose fine-grained errors into family errors and sibling errors. A family error occurs when the predicted class belongs to an incorrect basic-level family. A sibling error occurs when the prediction lies in the correct family but corresponds to the wrong fine-grained class.

Stage 1 reduces family-level errors, indicating improved coarse semantic structure. The full method further reduces sibling errors, suggesting that commitment-guided residual refinement improves within-family discrimination after reliable abstraction has been established.

F.18 Selection Ratio across Shot Groups

Table 32 reports the evaluation-time selection ratio across shot groups. The residual branch is activated only on a limited subset of samples and is concentrated on medium-shot and few-shot predictions.

The selection ratio is zero for many-shot predictions because the evaluation mask restricts residual correction to predicted medium-shot and few-shot classes. Among selected samples, the correction success rate is higher than the failure rate, supporting the usefulness of the commitment criterion.

Table 26: Sensitivity to evaluation commitment threshold τ_{eval} on CIFAR-100-LT (IR$=100$). Results are reported as mean $\pm$ standard deviation over repeated runs.

τ_{eval}	0.40	0.50	0.60	0.70	0.80
Selection ratio	0.42 ± 0.02	0.31 ± 0.02	0.24 ± 0.01	0.15 ± 0.01	0.07 ± 0.01
Overall accuracy	59.2 ± 0.5	60.0 ± 0.4	$\mathbf{60.6\pm0.4}$	60.1 ± 0.4	59.4 ± 0.5
Few-shot accuracy	36.2 ± 0.6	36.8 ± 0.5	$\mathbf{37.0\pm0.5}$	36.7 ± 0.6	36.1 ± 0.6

Table 27: Distillation variants in Stage 2 on CIFAR-100-LT (IR$=100$). Results are reported as mean $\pm$ standard deviation over repeated runs.

Distillation variant	Many	Medium	Few	All
No distillation	77.29 ± 0.45	61.20 ± 0.48	36.97 ± 0.62	59.56 ± 0.46
Fine-level only	78.26 ± 0.36	62.40 ± 0.42	$\mathbf{38.03\pm0.55}$	$\mathbf{60.64\pm0.40}$
Basic-level only	77.29 ± 0.44	61.20 ± 0.49	36.97 ± 0.63	59.56 ± 0.47
Fine + basic distillation	$\mathbf{78.46\pm0.34}$	$\mathbf{63.11\pm0.40}$	36.80 ± 0.50	60.59 ± 0.36

F.19 Summary of Ablation Findings

The extended ablations support the following conclusions.

- Basic-level abstraction improves the fine-grained predictor before residual refinement is introduced.
- Hierarchical consistency is most effective when the supervised basic-level branch is used as a stop-gradient abstraction target.
- Stage 2 refinement should be guided by a stable teacher-based commitment signal rather than applied uniformly.
- Medium-shot and few-shot targeting is a practical default that covers both scarcity regimes without broad many-shot activation.
- The residual branch should be family-masked, centered, and regularized so that it behaves as a within-family correction rather than a global classifier.
- The improvement cannot be explained only by additional training time or parameter count.
- Semantically meaningful taxonomies outperform random, shuffled, overly coarse, and overly fine alternatives.
- Family and sibling error decomposition supports the abstraction-to-differentiation interpretation.

Together, these results provide evidence that BASICLT works not merely by adding auxiliary supervision or model capacity, but by controlling the semantic granularity of discrimination under long-tailed scarcity.

G Extended Analysis and Visualization

This appendix provides additional analyses and visualizations to better understand how BASICLT behaves. While Appendix F focuses on controlled ablations, this section focuses on mechanism-oriented evidence. Specifically, we analyze whether basic-level abstraction is easier to establish than fine-grained discrimination under scarcity, whether Stage 1 improves coarse semantic consistency, how Stage 2 selects samples for residual refinement, and when residual correction succeeds or fails.

Unless otherwise specified, the analyses are conducted on CIFAR-100-LT with IR$=100$, which provides an official and balanced basic-level taxonomy. This setting allows us to decompose predictions into family-level and sibling-level behavior in a controlled manner.

Table 28: Teacher initialization and warmup design on CIFAR-100-LT (IR= 100). Results are reported as mean ± standard deviation over repeated runs.

Variant	Many	Medium	Few	All
Teacher init + warmup	78.46±0.34	**63.11±0.40**	**36.80±0.50**	**60.59±0.36**
Teacher init + no warmup	**78.69±0.36**	62.69±0.43	**36.80±0.52**	60.52±0.38
Random refinement init + warmup	66.57±0.72	50.97±0.68	20.10±0.85	47.17±0.62
Random refinement init + no warmup	69.49±0.65	53.14±0.61	24.20±0.78	50.18±0.58

Table 29: Uniform versus selective refinement across benchmarks. All rows are controlled variants of our method under the same training and evaluation protocol for each dataset. Results are reported as mean ± standard deviation over repeated runs when available. Δ denotes the gain of selective refinement over uniform refinement.

Dataset	Stage 1 only	Uniform refinement	Selective refinement	Δ
CIFAR-100-LT IR= 100	56.1±0.3	55.8±0.5	**60.6±0.4**	+4.8
CIFAR-100-LT IR= 50	61.0±0.3	60.4±0.5	**65.7±0.4**	+5.3
CIFAR-10-LT IR= 100	86.5±0.2	86.2±0.3	**87.8±0.3**	+1.6
ImageNet-LT	60.4±0.2	60.0±0.3	**63.2±0.3**	+3.2
iNaturalist 2018	75.8±0.2	75.5±0.3	**77.2±0.3**	+1.7
Places-LT	42.6±0.3	42.4±0.4	**44.5±0.4**	+2.1

G.1 Family-Level Reliability versus Fine-Grained Reliability

A central motivation of BASICLT is that family-level abstraction can be more reliable than fine-grained discrimination for low-frequency classes. To verify this behavior, we compare family-level accuracy and fine-grained accuracy across many-shot, medium-shot, and few-shot groups.

For a prediction over fine-grained classes, we compute the induced family prediction by applying the aggregation operator $\mathrm{Agg}(\cdot)$ and taking the most likely family. A prediction is considered family-correct if the predicted family matches the ground-truth basic-level label, even if the fine-grained class is incorrect.

Table 33 reports the comparison. Across all shot groups, family-level accuracy is higher than fine-grained accuracy. The gap is especially large for medium-shot and few-shot classes, indicating that coarse semantic recognition is often more reliable than sibling-level discrimination under scarcity.

This observation supports the abstraction-to-differentiation design. If family-level structure is more reliable than fine-grained prediction for rare classes, then it is reasonable to first learn and stabilize this structure before introducing additional sibling-level refinement.

G.2 Effect of Stage 1 on Coarse Semantic Consistency

Stage 1 is designed to improve coarse semantic structure through direct basic-level supervision and hierarchical consistency. To analyze this effect, we measure the agreement between two family-level predictions:

- the family prediction from the basic-level branch $u(x)$;
- the family prediction induced by the fine-grained logits $\mathrm{Agg}(s(x))$.

We define coarse agreement as

$$\mathrm{Agree}(x) = \mathbb{I}\left[\arg\max_b u_b(x) = \arg\max_b \mathrm{Agg}(s(x))_b\right]. \tag{93}$$

Higher agreement indicates that the explicit basic-level branch and the fine-grained predictor share a more coherent coarse semantic structure.

Table 34 shows that Stage 1 improves coarse agreement across all shot groups. The improvement is especially clear for medium-shot and few-shot classes, suggesting that the abstraction objective helps organize low-frequency classes at the family level.

Table 30: Taxonomy sensitivity on **CIFAR-100-LT** (IR$=100$). Results are reported as mean $\pm$ standard deviation over repeated runs.

Taxonomy	# Families	Many	Medium	Few	All
Random grouping	20	73.0 ± 0.7	57.1 ± 0.6	33.8 ± 0.8	56.2 ± 0.6
Shuffled semantic grouping	20	73.8 ± 0.6	58.0 ± 0.6	34.4 ± 0.7	57.0 ± 0.5
Noisy taxonomy	20	75.0 ± 0.5	59.2 ± 0.5	35.1 ± 0.7	58.3 ± 0.5
Coarser taxonomy	10	75.8 ± 0.5	59.6 ± 0.5	35.3 ± 0.6	58.8 ± 0.4
Finer taxonomy	40	76.4 ± 0.5	60.1 ± 0.5	35.9 ± 0.6	59.4 ± 0.4
Official superclass taxonomy	20	$\mathbf{78.4\pm0.3}$	$\mathbf{63.2\pm0.4}$	$\mathbf{37.0\pm0.5}$	$\mathbf{60.6\pm0.4}$

Table 31: Family and sibling error decomposition on **CIFAR-100-LT** (IR$=100$). Lower is better. Results are reported as mean $\pm$ standard deviation over repeated runs.

Variant	Family error	Sibling error	Total error
Base predictor	28.6 ± 0.5	18.8 ± 0.6	47.4 ± 0.4
Stage 1 full	24.1 ± 0.4	19.8 ± 0.6	43.9 ± 0.3
Stage 2 uniform refinement	24.5 ± 0.5	19.7 ± 0.6	44.2 ± 0.5
Full BasicLT	$\mathbf{21.8\pm0.4}$	$\mathbf{17.6\pm0.5}$	$\mathbf{39.4\pm0.4}$

These results indicate that Stage 1 does not merely add an auxiliary coarse classifier. It also improves the compatibility between the fine-grained predictor and the basic-level abstraction, which is necessary for reliable commitment-guided refinement in Stage 2.

G.3 Family Error and Sibling Error Decomposition

To further analyze the error structure, we decompose fine-grained errors into family errors and sibling errors.

A *family error* occurs when the predicted class belongs to an incorrect basic-level family:

$$g(\hat{y}) \neq g(y). \tag{94}$$

A *sibling error* occurs when the predicted class belongs to the correct family but is not the correct fine-grained class:

$$g(\hat{y}) = g(y), \qquad \hat{y} \neq y. \tag{95}$$

This decomposition is useful because Stage 1 is expected to reduce family-level errors, while Stage 2 is expected to reduce sibling-level errors among committed samples.

Table 35 reports the results. Stage 1 reduces family errors, indicating improved coarse semantic abstraction. The full method further reduces sibling errors, suggesting that commitment-guided residual refinement improves within-family discrimination after the coarse structure is stabilized.

Uniform refinement does not reduce the error as effectively as commitment-guided refinement. This is consistent with the view that residual correction should not be applied broadly when family-level commitment is unreliable.

G.4 Selection Dynamics during Stage 2

Stage 2 trains the residual branch only on samples selected by the teacher-based commitment rule. To understand how selective the training process is, we track the training-time selection ratio:

$$\rho_\tau^{\text{train}} = \frac{1}{N}\sum_{i=1}^{N} m_i^{\text{train}}. \tag{96}$$

Figure 3 shows the selection ratio over Stage 2 training. The ratio remains moderate throughout training, indicating that the residual branch is not trained as a universal second classifier. Instead, it

Table 32: Evaluation-time selection ratio across shot groups on **CIFAR-100-LT** (IR$=100$). Ratios are reported as fractions. Results are shown as mean $\pm$ standard deviation over repeated runs.

Group	Many	Medium	Few	Overall
Selection ratio	0.00 ± 0.00	0.31 ± 0.02	0.42 ± 0.03	0.24 ± 0.01
Correction success rate	-	0.58 ± 0.03	0.61 ± 0.04	0.59 ± 0.03
Correction failure rate	-	0.21 ± 0.02	0.24 ± 0.03	0.22 ± 0.02
No-change / neutral rate	-	0.21 ± 0.02	0.15 ± 0.02	0.19 ± 0.02

Table 33: Family-level accuracy versus fine-grained accuracy on **CIFAR-100-LT** (IR$=100$). Family-level prediction is more reliable than fine-grained prediction, especially for lower-frequency classes. Results are reported as mean $\pm$ standard deviation over repeated runs.

Prediction level	Many	Medium	Few	Overall
Fine-grained accuracy	69.5 ± 0.4	52.4 ± 0.5	30.2 ± 0.6	52.6 ± 0.4
Family-level accuracy	86.7 ± 0.3	75.8 ± 0.5	61.4 ± 0.7	75.1 ± 0.4
Accuracy gap	17.2 ± 0.5	23.4 ± 0.6	31.2 ± 0.8	22.5 ± 0.5

receives supervision only on samples whose family-level structure is reliable according to the frozen teacher and whose labels belong to the medium-shot or few-shot groups.

This behavior is important for the interpretation of BASICLT. The residual branch is not intended to replace the fine-grained predictor. It is trained as a conditional correction mechanism whose supervision is restricted by semantic commitment and class-frequency eligibility.

G.5 Selection Distribution across Shot Groups

At inference time, BASICLT activates residual correction only when the student satisfies the commitment criterion and the base prediction lies in the target medium-shot or few-shot set. Table 36 reports the selection ratio across shot groups.

The zero selection ratio for many-shot predictions follows from the evaluation-time eligibility rule, which restricts residual correction to predicted low-frequency classes. Among selected samples, the correction success rate is higher than the failure rate. This supports the role of the commitment criterion as a practical signal for refinement utility. This table reports the selection statistics under the extended analysis protocol used in Appendix G; Appendix H.4 additionally reports wrong-family commitment and residual safety under the base-predicted shot grouping used in the main mechanism figure.

G.6 Commitment Threshold and Coverage–Reliability Trade-Off

The evaluation threshold τ_{eval} controls how conservative the residual correction is. A lower threshold increases coverage by activating refinement on more samples, while a higher threshold improves reliability but reduces coverage.

Figure 4 visualizes this trade-off. As the threshold increases, the selection ratio decreases monotonically. Accuracy first improves and then declines, producing an intermediate optimum. This pattern indicates that both overly permissive and overly conservative refinement are suboptimal.

This analysis supports the use of semantic commitment as a control signal. The residual branch is most effective when it is activated on a subset that is sufficiently reliable but still large enough to provide useful correction opportunities.

G.7 Residual Correction Behavior

We further examine how residual correction changes predictions. For a selected sample, the residual branch can have three effects:

Table 34: Coarse agreement between the basic-level branch and the family-level distribution induced by the fine-grained predictor on **CIFAR-100-LT** (IR$=100$). Results are reported as mean $\pm$ standard deviation over repeated runs.

Variant	Many	Medium	Few	Overall
Base predictor + basic branch	82.4 ± 0.4	70.3 ± 0.5	54.8 ± 0.8	69.2 ± 0.5
Stage 1 full	87.1 ± 0.3	76.5 ± 0.5	62.7 ± 0.7	75.8 ± 0.4
Full BASICLT	88.5 ± 0.3	78.4 ± 0.4	64.1 ± 0.7	77.3 ± 0.4

Table 35: Family and sibling error decomposition on **CIFAR-100-LT** (IR$=100$). Lower is better. Results are reported as mean $\pm$ standard deviation over repeated runs.

Variant	Family error	Sibling error	Total error
Base predictor	28.6 ± 0.5	18.8 ± 0.6	47.4 ± 0.4
Stage 1 full	24.1 ± 0.4	19.8 ± 0.6	43.9 ± 0.3
Stage 2 uniform refinement	24.5 ± 0.5	19.7 ± 0.6	44.2 ± 0.5
Full BASICLT	$\mathbf{21.8\pm0.4}$	$\mathbf{17.6\pm0.5}$	$\mathbf{39.4\pm0.4}$

- **Successful correction:** the base prediction is incorrect and the final prediction becomes correct.
- **Failed correction:** the base prediction is correct and the final prediction becomes incorrect, or the residual changes an incorrect prediction to another incorrect class.
- **Neutral correction:** the residual is activated but does not change the final predicted class.

Table 37 reports the correction behavior on selected samples. Successful corrections are more frequent than failures, especially in few-shot classes. This suggests that the residual branch is useful when activated under reliable commitment.

Here, "failed correction" also includes an incorrect prediction changing to another incorrect class. It is therefore broader than the correct-to-incorrect harmful transition used in Appendix H.4. These outcome definitions should not be interchanged when attributing changes in Top-1 accuracy.

This behavior is consistent with the residual branch being a selective local correction rather than an unconditional classifier. The correction is not always beneficial, which explains why commitment-based activation is necessary.

G.8 Qualitative Examples

Figure 5 shows qualitative examples of residual correction. We group examples into three categories.

Successful within-family correction. In these cases, the base prediction already identifies the correct family but confuses the target class with a visually similar sibling. The residual branch corrects the prediction within the committed family. These examples illustrate the intended use of Stage 2: once the family-level structure is reliable, the residual branch refines sibling-level distinctions.

Unchanged prediction under low commitment. In these cases, the model does not activate residual correction because the family-level commitment criterion is not satisfied. The prediction therefore falls back to the base fine-grained logits. These examples show that BASICLT avoids applying residual correction when the coarse semantic structure is unreliable.

Failure cases. In some cases, the residual branch is activated but does not produce the correct class. Failures often occur when the family contains visually similar siblings, when the input is ambiguous, or when the base prediction lies in a plausible but incorrect low-frequency class. These cases highlight that commitment is a useful but imperfect proxy for refinement reliability.

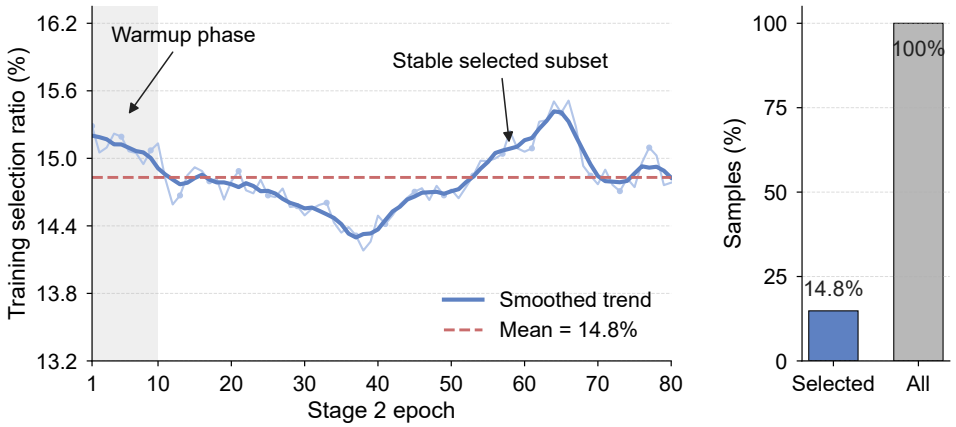


Figure 3: Training-time selection dynamics during Stage 2 on **CIFAR-100-LT** (IR= 100). The selection ratio remains low and stable across training, indicating that the residual branch is optimized on a limited subset of committed medium-shot and few-shot samples rather than on all samples. The right panel compares the average selected fraction with the full training set.

Table 36: Evaluation-time selection behavior across shot groups on **CIFAR-100-LT** (IR= 100). Ratios are reported as fractions. The residual branch is activated only on a limited subset of predictions and is concentrated on medium-shot and few-shot groups. Results are reported as mean ± standard deviation over repeated runs.

Group	Many	Medium	Few	Overall
Selection ratio	0.00±0.00	0.31±0.02	0.42±0.03	0.24±0.01
Correction success rate	-	0.58±0.03	0.61±0.04	0.59±0.03
Correction failure rate	-	0.21±0.02	0.24±0.03	0.22±0.02
No-change / neutral rate	-	0.21±0.02	0.15±0.02	0.19±0.02

The qualitative results support the interpretation that BASICLT performs conditional within-family differentiation. The most typical successful cases are not arbitrary class changes; they are corrections among semantically related sibling classes after family-level commitment has been established.

G.9 Visualization of Feature Organization

We visualize the learned representation before and after Stage 1 to examine whether basic-level abstraction improves coarse semantic organization. Figure 6 shows a two-dimensional embedding of validation features, colored by basic-level family and fine-grained class.

After Stage 1, samples from the same basic-level family become more coherent in the embedding space, while fine-grained classes remain distinguishable within each family. This visualization is consistent with the role of Stage 1: it does not collapse sibling classes into a single coarse cluster, but encourages a representation in which coarse family structure is more stable.

This visualization complements the quantitative coarse-agreement and error-decomposition results. It suggests that the basic-level branch shapes the representation in a way that supports later sibling-level refinement.

G.10 Analysis of Taxonomy Quality

The quality of the basic taxonomy affects both abstraction learning and selective refinement. We analyze this effect by comparing the official CIFAR-100 superclass taxonomy with random, shuffled, coarser, finer, and noisy alternatives, as reported in Appendix F.16.

Figure 7 visualizes the relationship between taxonomy quality and performance. Semantic taxonomies provide higher family-level accuracy and better final fine-grained accuracy than random or shuffled

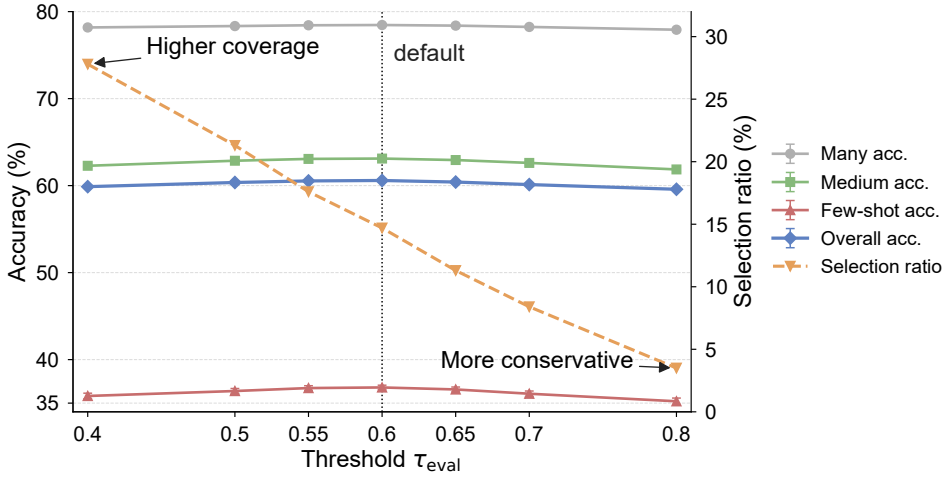


Figure 4: Coverage–reliability trade-off controlled by the evaluation commitment threshold on **CIFAR-100-LT** (IR= 100). Lower thresholds activate residual refinement more frequently but may include less reliable samples, while higher thresholds are more conservative and reduce the selected subset. The default threshold provides a balanced operating point across overall accuracy, shot-wise accuracy, and selection ratio.

Table 37: Residual correction behavior on selected medium-shot and few-shot samples on **CIFAR-100-LT** (IR= 100). Ratios are reported as fractions and sum to one within each group. Results are reported as mean ± standard deviation over repeated runs.

Group	Successful correction	Failed correction	Neutral correction
Medium-shot	0.58±0.03	0.21±0.02	0.21±0.02
Few-shot	0.61±0.04	0.24±0.03	0.15±0.02
Overall selected	0.59±0.03	0.22±0.02	0.19±0.02

groupings. Coarser taxonomies improve family-level confidence but reduce the usefulness of sibling-level refinement because each family becomes too broad. Finer taxonomies increase family specificity but weaken the abstraction effect because the family prediction becomes closer to the original fine-grained task.

This analysis reinforces the importance of moderate semantic granularity. The taxonomy should be coarse enough to provide shared support under scarcity, but specific enough to preserve meaningful sibling-level refinement.

G.11 Summary of Mechanism Analysis

The analyses in this appendix support the following interpretation of BASICLT.

- Family-level prediction is more reliable than fine-grained prediction under long-tailed scarcity, especially for medium-shot and few-shot classes.
- Stage 1 improves coarse semantic consistency between the basic-level branch and the family-level distribution induced by the fine-grained predictor.
- Stage 2 activates residual refinement only on a limited subset of committed low-frequency predictions, rather than using the residual branch as a universal classifier.
- The evaluation threshold controls a coverage–reliability trade-off, and an intermediate threshold provides the best performance.
- Residual correction succeeds most often when the base prediction is already within the correct semantic family but confuses sibling classes.

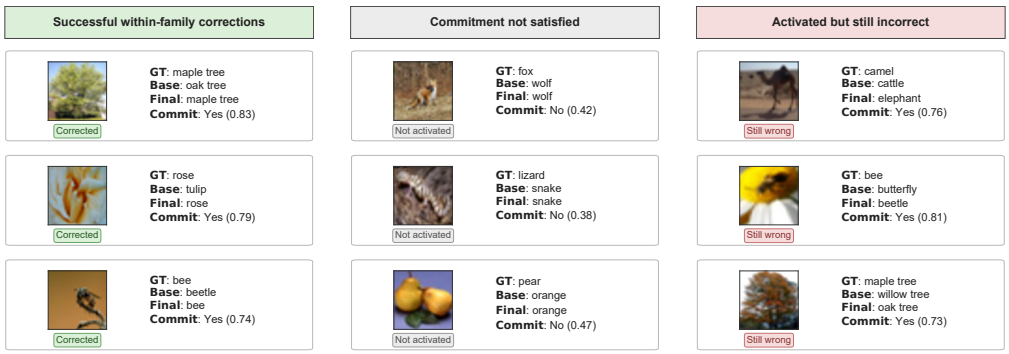


Figure 5: Qualitative examples of residual correction. Left: successful within-family corrections where the residual branch fixes the base prediction. Middle: examples where residual correction is not activated because the commitment criterion is not satisfied. Right: failure cases where residual correction is activated but still does not recover the correct fine-grained class.

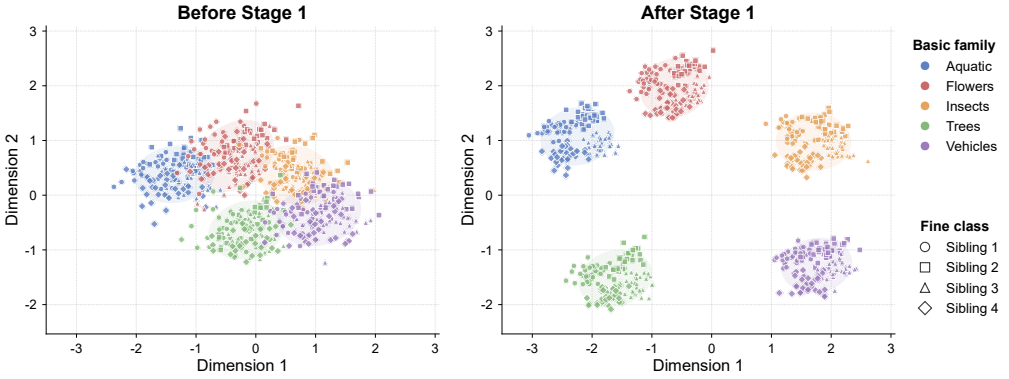


Figure 6: Feature visualization before and after Stage 1 on CIFAR-100-LT. We project real model features of the same sampled images into a shared two-dimensional space. Colors indicate basic-level families, and markers indicate fine-grained classes within each family. Stage 1 improves family-level organization while preserving within-family distinctions.

- Semantically meaningful taxonomies provide better abstraction and refinement behavior than random or poorly aligned groupings.

Together, these observations support the abstraction-to-differentiation view developed in the main paper. BASICLT first learns a stable basic-level abstraction and then uses family-level commitment to decide when additional fine-grained residual correction should be applied.

H Additional Controls for Taxonomy Fairness and Stage-Two Attribution

This appendix provides additional controls that directly address concerns about taxonomy usage, hierarchy-aware baselines, stage-two attribution, wrong-family commitment, and computational overhead. First, because BASICLT uses a fixed basic-level taxonomy, we clarify how the taxonomy is used and how it differs from changing the evaluation task. Second, we compare BASICLT against taxonomy-matched hierarchy-aware baselines that are allowed to use the same fine-to-basic mapping. Third, we decompose the gains in Stage 2 to separate the effects of additional training, teacher-student stabilization, residual refinement, and evaluation-time selective residual merging. Fourth, we analyze wrong-family commitment and residual safety at evaluation time. Finally, we report the practical training and inference overhead introduced by the two-stage framework.

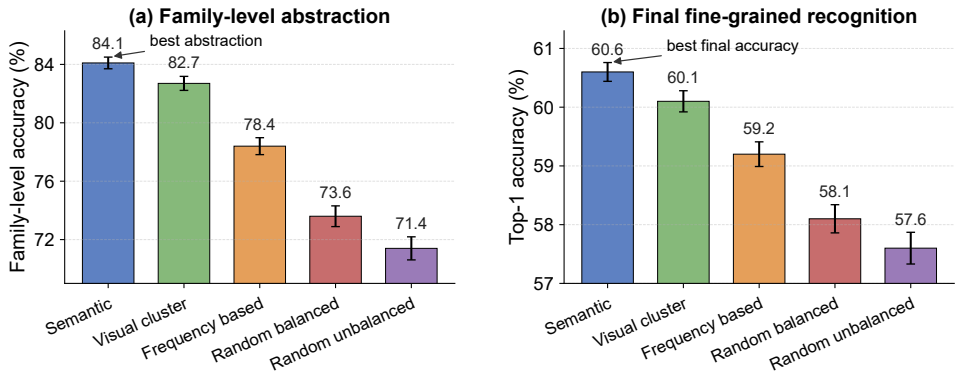


Figure 7: Effect of taxonomy construction on family-level accuracy and final fine-grained accuracy. Semantically meaningful taxonomies provide the best balance between abstraction and differentiation.

H.1 Taxonomy Usage and Fairness Protocol

BASICLT assumes a fixed mapping from fine-grained classes to basic-level families. This mapping is used only as an intermediate semantic scaffold for abstraction and selective refinement. It does not change the target label space, and all reported accuracies are evaluated on the original fine-grained classes. Importantly, the mapping is fixed before training and is not optimized using validation or test accuracy.

To make the role of taxonomy explicit, we separate three questions.

Does BASICLT change the evaluation task? No. The final task remains standard fine-grained classification over the original label space. The basic-level branch is not evaluated as the final classifier, and the residual branch only provides a conditional correction to the fine-grained logits.

Does BASICLT use taxonomy as an oracle at test time? No. At evaluation time, ground-truth family labels are unavailable. The residual correction is activated only from student-side predictions: the basic-level branch and the family distribution induced by the fine-grained logits must agree, and the base prediction must belong to the eligible medium/few-shot regime.

Is the gain merely caused by adding coarse labels? The results suggest that coarse labels alone do not explain the gain. The taxonomy provides a semantic scaffold, but the full improvement depends on how this scaffold is used: abstraction is learned first, and sibling-level residual refinement is introduced only under reliable family-level commitment. Random, shuffled, overly coarse, and overly fine taxonomies are weaker than the semantic taxonomy, and an auxiliary basic branch alone does not match the full method.

This protocol distinguishes BASICLT from a standard hierarchical classifier. Hierarchical supervision is not used as an alternative prediction tree. Instead, it defines the semantic level at which the model should first establish reliable abstraction before applying additional sibling-level differentiation.

H.2 Taxonomy-Matched Hierarchy-Aware Baselines

A natural concern is that BASICLT may benefit simply because it has access to a fine-to-basic mapping that most flat long-tailed baselines do not use. To address this concern, we construct taxonomy-matched baselines that are allowed to use the same mapping as BASICLT under the same backbone, training data, and augmentation. These controls are designed to isolate whether the gain comes from access to the taxonomy itself or from the proposed abstraction-to-differentiation mechanism.

We compare the following variants.

Flat base predictor. This is the underlying fine-grained model trained without basic-level abstraction or residual refinement.

Auxiliary basic branch. This variant adds the same basic-level branch as BASICLT but uses it only as an auxiliary multi-task loss. It tests whether coarse supervision alone is sufficient.

Table 38: Taxonomy-matched hierarchy-aware controls on CIFAR-100-LT with IR=100. All variants use the same fine-to-basic mapping when applicable. The comparison isolates whether the gain comes merely from access to taxonomy or from the proposed commitment-guided abstraction-to-differentiation design.

Variant	Many	Medium	Few	Overall
Flat base predictor	69.5±0.4	52.4±0.5	30.2±0.6	52.6±0.2
Auxiliary basic branch only	70.3±0.5	54.1±0.3	31.8±0.7	54.2±0.5
Hierarchical cross-entropy	71.0±0.4	55.2±0.5	32.7±0.6	55.1±0.4
Coarse-to-fine hard routing	67.8±0.6	55.7±0.5	33.1±0.6	54.7±0.5
Stage 1 full	72.1±0.3	56.4±0.6	33.5±0.6	56.1±0.4
Uniform within-family residual	70.4±0.5	57.0±0.4	34.2±0.3	55.8±0.2
BASICLT	**78.4±0.3**	**63.2±0.4**	**37.0±0.5**	**60.6±0.4**

Hierarchical cross-entropy. This variant augments the fine-grained objective with a family-level cross-entropy term computed from the aggregated fine-grained logits $\mathrm{Agg}(s(x))$. It therefore uses the same fine-to-basic mapping, but it does not introduce an explicit commitment mechanism or a family-conditioned residual correction branch.

Coarse-to-fine hard routing. This variant first predicts a family and then masks fine-grained logits outside the predicted family at inference time. It represents a conventional hierarchical prediction strategy. Unlike BASICLT, it routes predictions through a hard hierarchy and does not use a conservative fallback mechanism based on family-level commitment.

Uniform within-family residual refinement. This variant uses the same residual refinement branch as BASICLT but activates it without commitment-based selection. It tests whether the residual branch is useful when applied uniformly.

BASICLT. This is the full abstraction-to-differentiation framework with Stage 1 abstraction, teacher-stabilized commitment, and selective residual correction.

The taxonomy-matched comparison shows that access to basic-level labels alone is not sufficient. The auxiliary basic branch improves over the flat base predictor, indicating that coarse supervision is useful, but it remains far below the full framework. Hierarchical cross-entropy and coarse-to-fine hard routing also provide weaker results than BASICLT, suggesting that standard hierarchy-aware prediction does not fully address the long-tailed scarcity problem. Uniform residual refinement is weaker than commitment-guided refinement, confirming that the residual branch should not be activated indiscriminately. The strongest result is obtained only when basic-level abstraction is combined with teacher-stabilized, commitment-guided selective differentiation.

Although the residual branch is designed for medium/few-shot refinement, the many-shot gain should not be interpreted as unconditional residual activation on many-shot predictions. Under the inference rule, base predictions that already fall into the many-shot regime are not eligible for residual correction. Improvements on the many-shot group therefore mainly reflect the stabilized base pathway and the correction of eligible errors whose base predictions fall into medium/few-shot regimes, rather than direct residual refinement of confident many-shot predictions.

H.3 Strict Stage-Two Gain Decomposition

Stage 2 contains several design choices: continued optimization of the base pathway, teacher-student stabilization, residual refinement, and evaluation-time selective residual merging. To isolate these factors, we evaluate a strict decomposition where each row adds or removes one component while keeping the data, backbone, taxonomy, and Stage 2 training budget fixed.

The row "Stage 2 w/ distill., base pathway only" reports the student base pathway before applying the final evaluation-time residual merging. The full BASICLT row additionally applies the student-side commitment rule and merges the centered residual correction for eligible medium/few-shot predictions. This distinction separates the improvement of the stabilized base pathway from the final gain introduced by selective sibling-level correction.

Table 39: Strict Stage 2 gain decomposition on CIFAR-100-LT with IR=100. Except for the Stage 1-only reference, all Stage 2 variants start from the same Stage 1 checkpoint and use the same Stage 2 training budget. This table separates the effects of additional training, distillation, residual learning, and evaluation-time selective residual merging.

Variant	Many	Medium	Few	Overall
Stage 1 only	72.1±0.4	56.4±0.5	33.5±0.2	56.1±0.5
Stage 1 + continued base training only	72.8±0.4	56.9±0.5	33.8±0.4	56.6±0.4
Stage 1 + distillation only, no residual	74.1±0.3	58.5±0.4	34.3±0.5	57.4±0.4
Stage 1 + residual, no distillation	66.8±0.4	58.2±0.2	35.1±0.3	55.4±0.1
Stage 1 + distillation + uniform residual	73.6±0.4	59.4±0.3	35.2±0.4	57.7±0.3
Stage 2 w/ distill., base pathway only	75.3±0.2	60.8±0.4	35.8±0.5	58.7±0.5
Full BASICLT w/ selective residual merging	**78.4±0.3**	**63.2±0.4**	**37.0±0.5**	**60.6±0.4**

Table 40: Evaluation-time commitment reliability and residual safety on CIFAR-100-LT with IR=100. Rows are grouped by the base-predicted shot regime, matching the selection-ratio protocol used in the mechanism analysis. Selection ratio is computed over all evaluation samples in each base-predicted shot group. Correct-family commitment, wrong-family commitment, beneficial correction, and harmful correction are computed conditioned on selected samples. The residual branch is activated only on a limited committed subset. Wrong-family commitment remains controlled under the default threshold, and beneficial corrections outnumber harmful corrections.

Base-predicted group	Selection ratio	Correct-family commit.	Wrong-family commit.	Beneficial	Harmful	Selected-set net
Many	0.000±0.000	–	–	–	–	–
Medium	0.189±0.012	0.914±0.018	0.086±0.018	0.171±0.019	0.061±0.010	0.110±0.018
Few	0.270±0.018	0.872±0.026	0.128±0.026	0.203±0.027	0.081±0.017	0.122±0.026
Overall	0.147±0.010	0.902±0.019	0.098±0.019	0.181±0.021	0.069±0.012	0.112±0.020

The decomposition supports three conclusions. First, simply continuing Stage 2 training produces only limited gains and does not explain the final improvement. Second, residual learning without teacher-student stabilization improves some lower-frequency groups but can harm the overall head–tail balance, indicating that selective differentiation must preserve the Stage 1 abstraction. Third, stabilization and residual merging play different roles. Distillation improves the student base pathway by preserving family-level structure, while the final commitment-guided residual merging provides additional sibling-level correction only for eligible medium/few-shot predictions. Therefore, the full gain is not attributable to longer training, distillation alone, or residual capacity alone, but to their organization within the proposed abstraction-to-differentiation procedure.

H.4 Wrong-Family Commitment and Residual Safety

Because evaluation-time commitment is prediction-based, a possible failure mode is wrong-family commitment: the model may confidently commit to an incorrect family and then apply residual correction within the wrong sibling set. We therefore measure the reliability of selected samples and the effect of residual correction after selection.

For each evaluation sample, let $m_{\mathrm{eval}}(x)$ denote the selection mask and let $\hat{b}_u(x)$ be the committed family predicted by the student basic-level branch. We define the wrong-family commitment rate as

$$\mathrm{WCR} = \Pr\left[\hat{b}_u(x) \neq g(y) \mid m_{\mathrm{eval}}(x) = 1\right].$$

We also measure the correction outcome among selected samples. A selected correction is *beneficial* if the final prediction is correct while the base prediction is wrong, *harmful* if the base prediction is correct while the final prediction is wrong, and *unchanged* otherwise. The beneficial and harmful ratios in Table 40 are computed conditioned on selected samples.

This analysis clarifies why BASICLT uses conservative activation rather than unconditional residual merging. When the commitment criterion is satisfied, the selected subset has higher family-level reliability than the full evaluation set, and residual correction is more likely to improve than to harm the final prediction. When the commitment criterion is not satisfied, the model falls back to the base

Table 41: Computational overhead on CIFAR-100-LT with IR=100. Wall-clock cost is measured on a single NVIDIA RTX 4090 GPU. Peak memory is reported stage-wise to match the reproducibility details in Appendix D.6.

Configuration	Stage 1 time	Stage 2 time	Stage-wise peak memory	Test-time components
Base predictor	1.0–1.3 GPU-hours	–	Stage 1: ≈1.3 GB	Fine-grained predictor
BASICLT	1.0–1.3 GPU-hours	0.5–0.7 GPU-hours	Stage 1: ≈1.3 GB Stage 2: ≈0.6 GB	Base predictor + basic branch + conditional residual

fine-grained logits, preventing the residual branch from amplifying uncertain family-level decisions. The zero selection ratio for many-shot base predictions is expected because the evaluation-time eligibility rule targets medium/few-shot predictions, which are the regimes for which Stage 2 residual refinement is trained.

H.5 Computational Cost and Practical Overhead

BASICLT introduces a two-stage training pipeline, but the added components are lightweight relative to the underlying fine-grained predictor. Stage 1 adds a basic-level branch and hierarchical consistency loss. Stage 2 adds a teacher-stabilized residual refinement branch. The teacher is used only during training and is removed at inference time.

We report CIFAR-100-LT because it is the controlled setting used for most ablations; the same two-stage structure is used for the larger datasets, where the teacher remains training-only and is not used during inference. The memory range reports the peak across the two-stage configuration, while Appendix D.6 reports stage-wise memory measurements.

During Stage 2, the teacher forward pass is performed without gradient tracking, so the additional memory overhead is small compared with training two independent backbones. At inference time, BASICLT does not use the teacher. The additional operations consist of computing the basic-level branch, checking the commitment rule, and conditionally merging a centered residual within the committed family. If the commitment rule fails, the model directly falls back to the base fine-grained logits. Therefore, the main practical cost of BASICLT is the additional Stage 2 training, while inference remains close to the underlying fine-grained predictor.

H.6 Predicted Eligibility and Ground-Truth Shot Groups

Residual eligibility and reported shot-wise accuracy use different labels. The inference mask tests $\gamma(\hat{y}_{\text{base}}(x))$, whereas evaluation groups examples by $\gamma(y)$. A ground-truth many-shot example can therefore be selected when its base prediction is a medium-shot or few-shot class. A correctly classified many-shot example has a many-shot base prediction and is not selected. For the same checkpoint, the non-selected examples keep exactly the base logits.

The main component ablation reports many-shot accuracy of 72.1 ± 0.3 after Stage 1, 75.3 ± 0.2 for the stabilized Stage 2 base before residual merging, and 78.4 ± 0.3 after selective merging. Thus, both improved base predictions and correction of eligible errors contribute. The increase from the Stage 2 base to the final prediction is not evidence of unconditional refinement on many-shot base predictions.

Training also distinguishes sample eligibility from class competition. A committed medium/few-shot example is trained against all siblings in its family, which may include many-shot classes. Family masking restricts where logits are modified; it does not require the final predicted class to have the same frequency regime as the pre-merge prediction. In addition, centering is a logit constraint, not an exact preservation of family probability mass. The target-group comparison in Appendix F.8 measures the effect of extending refinement eligibility to all classes.

H.7 Summary of Additional Controls

The additional controls support the following conclusions.

- BASICLT does not change the target recognition task: all evaluation remains on the original fine-grained label space.
- Access to taxonomy alone does not explain the gain. Auxiliary basic supervision, hierarchical cross-entropy, hard coarse-to-fine routing, and uniform residual refinement are all weaker than commitment-guided abstraction-to-differentiation.
- Longer training alone is insufficient to match BASICLT, and the taxonomy-matched controls suggest that the gain is not explained by simply adding an auxiliary branch or a residual branch.
- Stage 2 gains are not merely caused by teacher-student distillation. Distillation stabilizes the Stage 1 structure and improves the student base pathway, while the selective residual branch provides conditional sibling-level refinement.
- Wrong-family commitment is an explicit failure mode, but conservative activation limits its impact by applying residual correction only on reliable committed samples and falling back to the base prediction otherwise.

I Limitations and Broader Impacts

This appendix briefly discusses the scope, limitations, and broader impacts of BASICLT.

I.1 Limitations

A fixed, single-membership taxonomy. BASICLT requires a meaningful intermediate partition specified before training. The current formulation does not support overlapping families or multi-inheritance, and it does not discover an appropriate taxonomy automatically. An ontology, metadata, or an audited visual grouping can supply a candidate partition, but constructing one can require domain knowledge. The taxonomy-matched controls isolate how a shared mapping is used; broad comparisons with flat methods do not remove the difference in available semantic information.

Taxonomy quality and coarse-family imbalance. Noisy, overly broad, overly fine, or visually inconsistent groups can weaken the abstraction signal and residual supervision. Unequal family sizes and training frequencies are allowed, but very sparse or heterogeneous families may themselves be difficult to recognize. The observed gains on nonuniform taxonomies do not guarantee robustness to arbitrary coarse-family imbalance. Singleton families admit no nonzero centered sibling correction. The fixed-seed visual-clustering analysis is limited evidence for one alternative construction, not a general taxonomy-discovery solution.

Commitment and distribution shift. Confidence and coarse agreement are imperfect proxies for refinement utility. The student can commit to the wrong family, and the true-label teacher selection used during training is not identical to label-free student selection during inference. A sample wrongly predicted as many-shot is excluded even when its true class is rare. The fallback preserves the base prediction when commitment fails, but does not certify its correctness or guarantee that an activated correction is harmless. Taxonomy perturbations do not substitute for evaluation under general domain shift.

Training complexity and evaluation scope. The procedure adds a frozen teacher, residual learning, loss schedules, and development-set operating-point selection. Warmup has only a small effect in the reported control, but this does not establish that all schedules can be removed simultaneously. The reported operating points also include dataset-specific thresholds. Detailed controlled studies concentrate on CIFAR-100-LT; cross-benchmark accuracy does not provide equally detailed evidence for every dataset. The extra Stage 2 computation is a real training cost, and selective merging alone is not a measured guarantee of lower inference cost.

I.2 Failure Cases

Representative failure cases follow naturally from the selective refinement design. First, if the model commits to an incorrect family, the within-family residual branch may refine the prediction in the wrong semantic region. Second, ambiguous, low-quality, or visually atypical samples may remain

difficult at both the family and sibling levels. Third, if the taxonomy provides weak sibling structure, residual refinement may offer limited benefit. These cases suggest that the reliability of family-level commitment and the quality of the basic-level grouping are important factors for applying BASICLT effectively.

I.3 Broader Impacts

Potential benefits. Long-tailed recognition is important in many real-world settings where rare categories matter. Improving recognition for medium-shot and few-shot classes can benefit applications such as biodiversity monitoring, medical image analysis, industrial inspection, remote sensing, and safety-critical perception, where rare categories may be important despite limited training data. By explicitly using intermediate semantic structure, BASICLT provides a way to improve rare-class recognition without relying solely on collecting more data.

Improved interpretability through semantic structure. The use of a basic-level taxonomy can make part of the model's behavior more interpretable. The model first estimates family-level structure and then applies residual refinement only under semantic commitment. This design allows analyses of family-level accuracy, sibling-level errors, selection ratios, and correction behavior. Such diagnostics can help practitioners understand when the model is relying on coarse abstraction and when it is applying fine-grained correction.

Risks from biased or inappropriate taxonomies. The same semantic structure that makes BASICLT interpretable can also introduce risks. A taxonomy may reflect human design choices, dataset biases, or culturally specific categories. If the taxonomy is inappropriate for the deployment setting, it may guide the model toward undesirable abstractions or reinforce biased groupings. For sensitive applications involving people or socially meaningful categories, taxonomy design should be handled carefully, and the use of fixed semantic groupings should be audited for potential harms.

Uneven benefits across classes. Although BASICLT is designed to improve recognition under scarcity, the benefit may not be uniform across all rare classes. Classes belonging to semantically coherent families may benefit more than classes in noisy, broad, or singleton families. As a result, performance improvements should be evaluated not only by overall accuracy, but also by per-class and per-group behavior. Deployment in high-stakes settings should include additional validation on the specific rare categories of interest.

Computational cost. The two-stage training pipeline increases training cost compared with a single-stage model. While the teacher is not used at inference and the residual branch is activated selectively, the training process still requires additional computation. This cost should be considered when applying the method to very large datasets or resource-constrained environments. Future work may explore more efficient training schedules, lightweight residual branches, or online taxonomy adaptation.

Responsible deployment. BASICLT is a general supervised recognition framework and does not remove the need for careful dataset curation, bias evaluation, calibration analysis, or human oversight. In applications where recognition errors can have serious consequences, the model should be evaluated under realistic distribution shifts and rare-class scenarios. The commitment mechanism provides a useful diagnostic signal, but it should not be interpreted as a guarantee of correctness.

Future directions. The limitations above suggest several directions for future work. One direction is to learn soft or adaptive taxonomies that better align semantic structure with visual similarity. Another is to extend the framework to settings where new classes arrive over time or where the taxonomy is uncertain. A third direction is to combine semantic granularity control with calibration, uncertainty estimation, or open-set recognition, so that the model can better decide when to refine, when to abstain, and when to request additional supervision.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We list our contributions in detail at the end of the Introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations in Appendix.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We provide the relevant assumptions and proofs in Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the code source in the abstract and the supplementary boundary in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide the source of the dataset and code in Appendix.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We describe the relevant details in Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Main results and core ablations for BASICLT are reported as mean $\pm$ standard deviation over five independent runs, and the tables explicitly indicate the uncertainty ranges. The reported variability corresponds to independent training runs with different random seeds.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We mention the computational resources used for the experiments in Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have read and understood the NeurIPS Code of Ethics and have done our best to conform to it. The work uses public supervised image-classification benchmarks and does not involve human subjects, scraped sensitive data, or high-risk generative models.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix H.3 discusses potential benefits for rare-category recognition and interpretability, as well as potential risks from biased or inappropriate taxonomies, uneven benefits across classes, computational cost, and the need for responsible deployment in high-stakes settings.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release high-risk generative models, scraped datasets, or other assets with substantial misuse risk. It uses public supervised image-classification benchmarks and releases only standard training/evaluation code and taxonomy mappings.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use public benchmark datasets and cite their original sources in Section 4.1 and the references. We do not redistribute raw datasets; users are instructed to obtain them from the official sources and comply with the corresponding licenses and terms of use. Any released code will include its own license and list required open-source dependencies.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We will release our code base with included readme files. We do not ship any source code or binary files from any other existing libraries.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This work does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This work does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: Our work does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.