Journal of Creative Student Research
Volume. 3, Nomor. 6 Desember 2025
e-ISSN: 2963-4776; dan p-ISSN: 2963-5942, Hal. 140-165
DOI: https://doi.org/10.55606/jcsr-politama.v3i6.6154
Tersedia: https://ejurnal.politeknikpratama.ac.id/index.php/jcsr

Analysis of EfficientNet-B0 with Sample Reweighting and Early-Learning Regularization for Food Recognition under Label Noise

Danang[1*], Toni Wijanarko Adi Putra[2]
[1-2] Universitas Sains dan Teknologi Komputer, Indonesia
Email: danang150787@gmail.com [1], toni.wijanarko@stekom.ac.id [2]
**Corresponding Author: danang150787@gmail.com*

Abstract. Food recognition systems are commonly developed under the assumption that training labels are fully accurate. In real-world applications, however, food image datasets frequently contain noisy annotations caused by incorrect user inputs, weak labeling mechanisms, or automated data collection processes. This study investigates the robustness of supervised food recognition under synthetic label noise using the Food-101 dataset Food-101. The research employs EfficientNet-B0 as a computationally efficient backbone model and compares conventional cross-entropy learning with a robust training approach that integrates two mechanisms: (1) small-loss sample reweighting to reduce the influence of potentially corrupted samples, and (2) an early-learning stopping strategy based on the memorization gap between noisy training accuracy and clean validation accuracy. Symmetric label noise levels of 20% and 40% are introduced only into the training data, while validation and testing datasets remain unaffected. Experimental results on a 20-class subset demonstrate that the proposed approach substantially improves clean test accuracy from 0.6476 to 0.8176 under 20% noise and from 0.5636 to 0.6928 under 40% noise. In addition, probability calibration performance measured by Expected Calibration Error (ECE) is improved from 0.1474 to 0.0813 at the 40% noise setting. Additional experiments on the complete 101-class dataset also reveal consistent performance improvements despite shorter training durations. The findings suggest that combining loss-aware sample weighting with memorization-aware early stopping can provide an efficient and practical solution for building robust and reliable food recognition models in noisy-label environments.

Keywords: EfficientNet-B0; Food Recognition; Label Noise; Probability Calibration; Sample Reweighting.

1. INTRODUCTION

Food image classification supports practical workflows such as dietary logging, restaurant discovery, and health-oriented applications where users expect fast and accurate recognition from photos. Despite this usefulness, the labels encountered in real deployments are often imperfect. Users may select the wrong class due to ambiguity (e.g., visually similar dishes or mixed plates), crowdsourced annotations can be inconsistent across annotators and contexts, and weak supervision or semi-automatic labeling pipelines can introduce systematic errors that persist across many samples. In addition, food categories can be inherently ambiguous because a single photo may contain multiple ingredients, multiple dishes, or non-standard presentation styles, all of which increase the likelihood of label mismatch between the image content and the assigned class.

From a learning perspective, label noise is not merely a minor nuisance: it can fundamentally distort the gradient signal and steer training toward spurious decision rules. Under corrupted supervision, empirical risk minimization may overweight mislabeled examples, encouraging the 1 model to fit artifacts that do not generalize. The failure mode is

Naskah Masuk: 08 Oktober 2025; Revisi: 17 November 2025; Diterima: 26 Desember 2025;
Terbit: 30 Desember 2025

especially pronounced in high capacity deep networks that can eventually fit almost any labeling pattern when optimization proceeds long enough.

A key empirical observation is that deep networks often exhibit a two-phase behavior under noise. They typically fit clean and easy patterns early in training, achieving rapid improvements in validation accuracy, but later they can drift into memorizing noisy labels, which degrades generalization (Arpit et al., 2017; Liu et al., 2020). This late-stage memorization is doubly problematic: it not only reduces test accuracy but also tends to produce overconfident predictions, which is undesirable in user-facing settings where probabilities are interpreted as confidence. Therefore, robustness to label noise should be evaluated not only by accuracy but also by reliability indicators that reflect how trustworthy the predicted probabilities are when training supervision is corrupted.

These considerations motivate training procedures that explicitly manage noisy supervision while remaining practical under realistic compute constraints. Instead of relying on complex multi-network systems or heavy noise modeling, we aim for a lightweight strategy that can be executed on limited resources (e.g., a single T4 GPU) and that directly targets the two dominant risks: (1) the undue influence of mislabeled samples during gradient updates, and (2) late-stage memorization that erodes generalization and calibration.

We study food recognition when a fraction of training labels is corrupted. Starting from Food-101 (Bossard et al., 2014), we create a controlled robustness benchmark by injecting symmetric label noise into the training split while keeping validation and test splits clean. This setting is intentionally chosen to be reproducible and interpretable: by holding evaluation splits clean, changes in accuracy and calibration can be attributed to the learning procedure under noisy supervision rather than to contamination of evaluation labels. Symmetric noise, while simplified, provides a standardized stress test that is widely used to compare robustness methods under increasing corruption rates, and it allows us to measure how quickly performance degrades as noise increases.

Within this benchmark, our objective is to quantify how a strong, compute-efficient baseline behaves under label corruption and to determine whether a single-model training recipe can substantially improve robustness without architectural changes. EfficientNet-B0 is a sensible testbed because it provides competitive accuracy with favorable efficiency, making it common in practical pipelines where compute and iteration time matter. The first goal is to establish baseline robustness under 0%, 20%, and 40% symmetric noise, so that the cost of noisy supervision is visible in a consistent experimental protocol.

The second goal is methodological: we seek a compute-feasible strategy that reduces the influence of likely-noisy samples during training and limits late-stage memorization. Rather than attempting to perfectly identify noisy labels, we leverage training dynamics-specifically the tendency of networks to assign lower loss to clean/easy samples earlier-to bias learning toward data that are more likely to be consistent. In parallel, we incorporate an early-learning control criterion to stop training when signals of memorization begin to dominate. The third goal is evaluative and diagnostic: beyond test accuracy, we examine calibration behavior (including ECE), runtime overhead relative to standard training, and qualitative evidence such as noisy-label candidates and representative failure cases. This broader view is necessary because a method that marginally improves accuracy but yields systematically overconfident probabilities, or that is prohibitively slow, is less useful in practice.

This work makes three main contributions aimed at bridging robustness research with practical constraints. First, we introduce a lightweight robust training recipe for EfficientNet-B0 that is explicitly designed to operate under a single-model, limited-compute setting while addressing noisy-label failure modes. The recipe combines a small-loss driven mechanism that prioritizes samples that remain consistent with the model's early learning signal, together with an early learning stopping criterion based on a memorization-gap indicator. The design is motivated by the empirically observed progression from fitting clean patterns to memorizing noise in deep networks (Arpit et al., 2017; Liu et al., 2020). By coupling selective emphasis of likely-clean data with a principled stopping signal, the method aims to improve generalization under corruption without requiring multi-network co-training, sophisticated semi-supervised pipelines, or explicit noise-transition estimation. This positioning also aligns with broader evidence that hybrid deep-learning designs can improve system robustness when the added control logic is kept lightweight and operationally targeted.

Second, we provide a focused empirical study on Food-101 under controlled symmetric noise at 20% and 40% (Bossard et al., 2014). We report not only accuracy but also probability reliability through calibration diagnostics such as ECE, because in many downstream scenarios the confidence score is operationally important (e.g., deciding whether to ask the user for confirmation, trigger a fallback, or abstain). We additionally quantify runtime overhead so that robustness improvements can be interpreted in the context of practical training budgets. Finally, we include qualitative diagnostics noisy-label candidate inspection and failure-case analysis to make the observed behavior more interpretable and to provide evidence that the method is addressing label corruption rather than merely shifting errors.

Third, we prioritize auditability and reproducibility by producing a complete artifact bundle that supports verification and downstream reuse. The bundle is designed to include paper ready figures and tables, per-experiment summaries, representative predictions for inspection, and checkpoints, enabling straightforward auditing of reported results and facilitating future extensions (e.g., different noise rates, alternative schedules, or additional baselines). This artifact-first emphasis is particularly important in label-noise studies, where subtle changes in data corruption procedures, split construction, or early-stopping behavior can materially affect outcomes; providing structured artifacts helps ensure that revisions and follow-up experiments remain traceable and consistent.

2. LITERATURE REVIEW

Food Recognition and EfficientNet

Food recognition has been studied extensively as a challenging fine-grained visual classification problem due to high intra-class variation (e.g., the same dish presented in different lighting, angles, or plating styles) and inter-class similarity (e.g., different dishes sharing similar textures or ingredients). Food-101 Bossard et al. (2014) is widely adopted as a standard benchmark because it provides a large number of categories with substantial diversity per class, making it useful for evaluating both representation quality and generalization. In practice, Food-101 is also often used as a proxy for realistic scenarios where label ambiguity exists, which makes it relevant for studying robustness when training supervision is imperfect.

From the model perspective, modern food recognition pipelines typically rely on convolutional backbones that have proven effective on large-scale visual recognition. Architectures such as ResNet He et al. (2016) and Inception-style designs Szegedy et al. (2016) established strong baselines for image classification, and large-scale pretraining on ImageNet has become a standard recipe to improve data efficiency and representation quality (Russakovsky et al., 2015). EfficientNet extends this lineage by systematically rethinking model scaling (depth, 3 width, and resolution) to achieve a strong accuracy efficiency trade-off (Tan & Le, 2019). This trade-off matters for noisy-label studies because robustness experiments often require multiple runs (noise levels, schedules, stopping rules, and ablations), and the computational budget can quickly become a limiting factor. Consequently, EfficientNet-B0 is a practical backbone for a robustness benchmark: it is strong enough to be meaningful, yet efficient enough to allow repeated experimentation under constrained hardware.

Although our primary focus is label noise, it is worth noting that robustness can be stressed from different angles. Common corruption benchmarks evaluate performance under distribution shifts in input appearance (Hendrycks & Dietterich, 2019). In contrast, label-noise robustness focuses on corrupted supervision rather than corrupted inputs. Both perspectives are comple mentary: a model can be robust to input corruptions yet brittle to training noise, or vice versa. This work specifically isolates the supervision aspect by injecting controlled label corruption into training while keeping evaluation splits clean, which allows us to study training dynamics and mitigation strategies in a reproducible manner.

Learning with Noisy Labels

Learning under noisy supervision is a well-established problem, and a comprehensive overview highlights diverse strategies ranging from explicit noise modeling to implicit regularization (Song et al., 2022). A key empirical phenomenon underpinning many practical methods is that deep networks often learn clean and easy patterns early and progressively memorize noisy labels later, which harms generalization (Arpit et al., 2017; Liu et al., 2020). This training-dynamics view is particularly important for compute-feasible methods because it enables heuristics based on observable signals (loss curves, confidence, or validation behavior) without requiring expensive auxiliary models.

Loss correction and robust losses. One classical direction is to correct the loss by modeling the noise transition matrix, thereby connecting observed noisy labels to latent clean labels (Patrini et al., 2017). When the transition structure is well-estimated, loss correction can provide principled improvements; however, estimation itself can be brittle under severe corruption or limited clean validation, and it may introduce additional modeling and tuning complexity. Another family replaces the standard cross-entropy objective with robust alternatives that reduce sensitivity to outliers or mislabeled samples. Generalized cross entropy is a representative example designed to interpolate between objectives and mitigate the influence of noisy labels (Zhang & Sabuncu, 2018). Robust-loss approaches often have attractive simplicity but can require careful hyperparameter selection to avoid underfitting clean hard examples, especially in f ine-grained classification.

Curriculum learning, sample selection, and reweighting. A second major line leverages curriculum signals such as the “small-loss” effect, where clean samples tend to have lower losses earlier in training. MentorNet learns a data-driven curriculum by training a teacher-like module that guides the student network toward more reliable samples (Jiang et al., 2018). While effective, such approaches may increase system complexity by introducing additional networks or learned heuristics. Example reweighting offers a simpler perspective: instead of discarding

data, one can assign weights to training examples so that the model focuses more on samples that contribute positively to generalization (Ren et al., 2018). In noisy-label settings, reweighting can be used as an implicit filter, but it must balance two competing goals: suppressing truly mislabeled samples while retaining informative hard-but-correct samples that are essential for fine-grained recognition. This trade-off is especially relevant for food images, where visually difficult examples can be correct yet receive high loss early in training.

Two-network and semi-supervised style strategies. Co-Teaching trains two networks simultaneously and has each network select small-loss samples for the other, which reduces confirmation bias by decoupling selection from fitting (Han et al., 2018). This idea is influential because it explicitly addresses a common failure mode of single-network selection: the model may reinforce its own mistakes by repeatedly selecting samples consistent with its current biases. Building on these concepts, more complex approaches combine mixture modeling and semi-supervised learning to separate clean and noisy subsets and exploit unlabeled-like objectives; DivideMix is a prominent example (Li et al., 2020). Unsupervised noise modeling methods also attempt to learn structure in noisy labels without direct clean supervision (Arazo et al., 2019). While these approaches can achieve strong robustness, they often involve iterative procedures, multiple models, or additional stages (e.g., warmup, estimation, SSL training), which increases computational cost and implementation burden. For many practical projects or academic settings with limited hardware, these requirements can be prohibitive.

Early-learning regularization and training-dynamics control. A complementary direction directly targets memorization behavior. Early-learning regularization (ELR) is designed to prevent late-stage memorization of noisy labels by stabilizing predictions and discouraging drift toward fitting corrupted supervision (Liu et al., 2020). This connects closely to the broader observation that the timing of optimization matters: stopping too late can be harmful under noise even if training accuracy continues to improve. Related techniques that shape the training signal include label smoothing, which can reduce overconfidence and sometimes improve generalization (Müller et al., 2019), and abstention-based learning, which allows the model to defer predictions for uncertain cases and can be helpful in noisy regimes (Thulasidasan et al., 2019). These methods highlight that robustness can be improved not only by identifying noisy samples, but also by controlling how confidently and how long the model is allowed to fit the training signal.

lso by controlling how confidently and how long the model is allowed to fit the training signal. Overall, the literature suggests a spectrum: at one end, highly engineered pipelines can achieve strong robustness but are heavy; at the other end, lightweight heuristics that exploit

early-learning signals can be attractive if they deliver measurable gains with minimal overhead. This paper positions itself closer to the latter, while still grounding design choices in established training-dynamics observations (Arpit et al., 2017; Liu et al., 2020) and example weighting principles (Ren et al., 2018).

Calibration Under Noise

Calibration concerns whether predicted probabilities reflect true correctness likelihood. Even highly accurate neural networks can be miscalibrated, often exhibiting overconfidence (Guo et al., 2017). In noisy-label regimes, calibration becomes more delicate for two reasons. First, corrupted supervision can distort the mapping between confidence and correctness: the model may become confident about patterns that correlate with noisy labels rather than with the true class. Second, techniques that improve robustness by filtering or reweighting samples can change the effective training distribution, which may either improve reliability (by reducing noisy gradients) or worsen it (by encouraging overly conservative fitting), depending on the mechanism.

We evaluate calibration using reliability diagrams and Expected Calibration Error (ECE) computed via binning, a widely used diagnostic family (Guo et al., 2017). Binning-based calibration analysis is closely related to Bayesian binning and related histogram-based approaches that assess probability quality by grouping predictions into confidence bins and comparing predicted confidence with empirical accuracy (Naeini et al., 2015). Calibration is also connected to uncertainty estimation perspectives that aim to produce more trustworthy probabilistic outputs (Kuleshov et al., 2018). In practical applications, better calibration can be operationally valuable: it can support thresholding, abstention policies, or human-in-the-loop confirmation f lows. Therefore, we treat calibration as a first-class evaluation target alongside accuracy, rather than as an afterthought.

Gap Addressed

Although state-of-the-art methods for learning with noisy labels can be highly effective, many of them are computationally expensive or operationally complex. Two-network methods require training multiple models concurrently (Han et al., 2018), while semi-supervised or mixture-model based pipelines introduce iterative stages and additional components that increase both runtime and engineering overhead (Arazo et al., 2019; Li et al., 2020). Loss correction approaches can demand accurate transition estimation and careful tuning (Patrini et al., 2017), and robust-loss alternatives may introduce additional hyperparameters that interact with dataset difficulty and noise severity (Zhang & Sabuncu, 2018). As a result, there remains

a practical need for approaches that are single-model, compute-feasible, and still deliver measurable gains under substantial noise.

This work targets that gap by emphasizing a lightweight recipe grounded in observed training dynamics under label corruption. We leverage the early-learning behavior of deep networks (Arpit et al., 2017; Liu et al., 2020) to reduce the influence of likely-noisy examples via small-loss driven emphasis and to control training duration via an early-learning stopping signal. Importantly, the goal is not to compete with the most complex pipelines on absolute peak robustness, but to provide a robust and auditable improvement that can be executed under limited compute budgets while also improving probability reliability as measured by calibration diagnostics (Guo et al., 2017; Naeini et al., 2015).

3. PROPOSED METHOD

Problem setup

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ denote the training set, where xi is an RGB food image and $y_i \in \{1, \dots, C\}$ is its class label. We train a parametric classifier fθ that outputs a categorical distribution over C classes, i.e., $p_\theta(\cdot \mid x)$ = softmax $\big(f_\theta(x)\big)$. In this study, we explicitly separate the roles of the data splits to obtain an unambiguous robustness evaluation: only the training labels are corrupted by synthetic noise, while validation and test labels remain clean. This design ensures that performance differences can be attributed to the training procedure under noisy supervision rather than to contamination of the evaluation protocol.

Concretely, training uses corrupted labels ˜yi (defined below), and model selection is performed using a clean validation set to choose stopping time and the best checkpoint. Final reporting is conducted on a clean test set. This setup reflects a realistic scenario where training data may be noisy, but a smaller trusted set (validation) can still be available for model selection, and the test set represents true performance under clean ground truth.

Synthetic Symmetric Label Noise

To obtain a controlled and reproducible noisy-label benchmark, we generate symmetric (uniform) label corruption on the training split. For a target noise rate r ∈ [0,1], each training label is independently corrupted as follows: with probability 1 −r the label is kept unchanged, and with probability r it is replaced by a uniformly sampled incorrect class drawn from the remaining C −1classes. Validation and test labels remain clean to ensure valid robustness measurement.

This corruption model is intentionally simple: it imposes no class-dependent structure and therefore provides a standardized stress test for robustness comparisons across methods (Song et al., 2022). In expectation, a fraction r of training labels becomes incorrect, and the induced noise transition is uniform among the remaining classes. Importantly, the validation and test labels remain clean to preserve the integrity of model selection and final evaluation. As a result, improvements observed under higher r can be interpreted as genuine robustness gains against corrupted supervision rather than artifacts of evaluation noise.

In practical terms, we implement label corruption as a deterministic mapping given a fixed random seed, so that experiments at each noise level can be reproduced exactly. This is particularly important when comparing different training strategies, since small differences in which samples are corrupted can otherwise introduce variance that obscures methodological effects.

Baseline: EfficientNet-B0 with Cross-Entropy

As a strong and compute-efficient baseline, we adopt EfficientNet-B0 (Tan & Le, 2019) initialized with ImageNet pretraining (Russakovsky et al., 2015). EfficientNet-B0 provides a favorable accuracy–efficiency trade-off, allowing multiple robustness runs (noise levels, schedules, and ablations) within constrained compute budgets.

The baseline optimizes the standard cross-entropy loss using corrupted training labels ˜yi:

$$\mathcal{L}_{\mathrm{C}}(i) = -\log p_{\theta}\left(\tilde{y}_i \mid x_i\right), \quad (1)$$

Where $p_{\theta}(\cdot \mid x)$ denotes the softmax output of the model. Over a minibatch B, the training objective is the average loss $\frac{1}{|\mathcal{B}|}\sum_{i\in\mathcal{B}}\mathcal{L}_{\mathrm{C}}(i)$, optimized via standard stochastic gradient methods. Under label noise, this baseline is expected to degrade as r increases because the gradients corre sponding to corrupted labels become systematically misleading. Establishing this degradation curve is necessary to quantify how much robustness a proposed method recovers relative to vanilla training.

Proposed: Small-Loss Reweighting + Early-Learning Regularization

Our method targets two empirically observed failure modes in noisy-label learning: (i) corrupted labels inject harmful gradients that can dominate learning if treated equally, and (ii) late-stage training can shift from learning true structure to memorizing noise, reducing generalization. The proposed approach addresses both with a single-model procedure: first, it reduces the influence of high-loss (potentially noisy) samples through small-loss driven selection; second, it controls training duration using an early-learning criterion based on a

memorization-gap signal. The design is grounded in the observation that deep networks tend to fit clean patterns early and memorize noise later (Arpit et al., 2017; Liu et al., 2020), and in prior work that leverages small-loss behavior for sample selection (Han et al., 2018; Jiang et al., 2018).

Small-loss sample reweighting. At each epoch t, we compute per-sample cross-entropy losses $\mathcal{L}_{\mathrm{C}}(i)$ over the training set (or an equivalent per-sample estimate accumulated across minibatches). We then select a low-loss subset of size $k = \lfloor rr(t) \cdot N \rfloor$, where rr(t) is a retain-rate schedule. Intuitively, under noisy supervision, samples with smaller losses are more likely to be correctly labeled in the early learning phase, whereas persistently high-loss samples have a higher chance of being corrupted or atypical (Han et al., 2018; Jiang et al., 2018). The key point is not that small-loss samples are always clean, but that the loss ranking provides a useful signal to bias training toward more reliable supervision when noise is present. We implement hard selection via a Top-k operator:

$$\mathcal{S}_t = \mathrm{TopKSmallLoss}(\{\mathcal{L}_{\mathrm{C}}(i)\}_{i=1}^{N}, k). \quad (2)$$

Training updates at epoch t use only samples in St, which is equivalent to assigning binary weights $w_i \in \{0,1\}$ such that $w_i = 1$ if $i \in S_t$ and $w_i = 0$ otherwise. This yields an effective objective $\sum_{i\in\mathcal{B}} w_i \mathcal{L}_{\mathrm{C}}(i)$ per minibatch. In contrast to two-network selection strategies that exchange subsets (Han et al., 2018), our approach remains single-model and therefore more compute-feasible. The retain-rate schedule rr(t) can be interpreted as a curriculum over the training set: it controls how conservative the method is in trusting the current loss signal, and it allows the procedure to focus on the most consistent samples when needed.

Early-learning control via memorization gap. Small-loss selection alone can reduce the influence of likely-noisy samples, but it does not fully eliminate the risk of late-stage memorization, especially when training continues after validation performance saturates. We therefore incorporate an early-learning control criterion based on a memorization-gap signal:

$$g(t) = \mathrm{Acc}_{\mathrm{train}}^{\mathrm{noisy}}(t) - \mathrm{Acc}_{\mathrm{val}}^{\mathrm{clean}}(t) \quad (3)$$

Here, $\mathrm{Acc}_{\mathrm{train}}^{\mathrm{noisy}}(\mathrm{t})$ measures training accuracy computed against corrupted labels, while $\mathrm{Acc}_{\mathrm{val}}^{\mathrm{clean}}(\mathrm{t})$ measures generalization on the clean validation set. Under label noise, it is common to observe that training accuracy can keep increasing even when validation accuracy stagnates or degrades; this divergence is a practical indicator that optimization is increasingly fitting the corrupted supervision rather than improving true generalization (Arpit et al., 2017; Liu et al., 2020).

Operationally, we monitor validation accuracy and the trend of g(t). When validation accuracy shows no meaningful improvement for several epochs (a patience window) while the gap g(t) continues to increase, we stop training and retain the best validation checkpoint. This implements a lightweight form of early-learning regularization: it does not introduce an additional penalty term, but it explicitly constrains training time to remain within the regime where learning is dominated by clean signal rather than memorization (Liu et al., 2020). The criterion is intentionally simple, auditable, and compatible with limited compute.

Mechanism Visualization

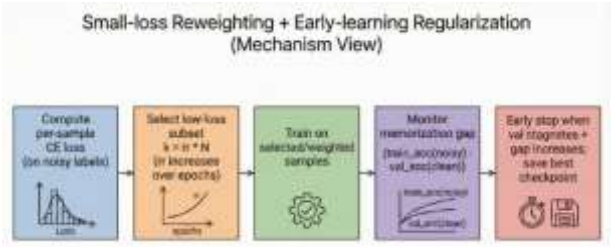

Figure 1. Mechanism view: small-loss selection reduces the influence of high-loss (likely noisy) samples, while early stopping uses a memorization gap to prevent late-stage overfitting to noise.

Figure 1 summarizes the proposed training mechanism. The diagram emphasizes two interacting effects. First, small-loss selection reduces the gradient contribution from high-loss samples, which are more likely to be mislabeled or difficult outliers under noisy supervision (Han et al., 2018; Jiang et al., 2018). Second, the memorization-gap monitor provides a training-dynamics lens: when noisy training accuracy continues to rise without corresponding validation improvement, continued optimization is more likely to reinforce noise-fitting behavior that harms generalization (Arpit et al., 2017; Liu et al., 2020). Together, these components yield a single-model procedure that aims to preserve early generalization gains while avoiding late-stage overfitting to corrupted labels.

End-to-end Pipeline

Figure 2. Training and evaluation pipeline on Food-101. Noise is injected into train only (validation/test remain clean). We compare a baseline CE model with the proposed reweighting + early-learning approach and export all paper artifacts.

Figure 2 outlines the full experimental pipeline used in this study. We begin by constructing train/validation/test splits and injecting synthetic symmetric noise into the training labels only, producing a controlled noisy-supervision environment. We then train (i) a baseline EfficientNet B0 with standard cross-entropy and (ii) the proposed small-loss plus early-learning procedure, using the same backbone and comparable optimization settings so that differences are attributable to the robustness mechanism rather than confounding factors.

Beyond training and evaluation, the pipeline explicitly includes artifact export for auditability and reproducibility: we save paper-ready tables and figures, record per-epoch metrics (train ing/validation accuracy and calibration statistics), and retain the best validation checkpoint for each setting. Implementation is carried out in PyTorch Paszke et al. (2019) with dataset access and split handling via the datasets library (Lhoest et al., 2021). Optimization uses Adam-style methods Kingma & Ba (2015) and decoupled weight decay when applicable Loshchilov & Hutter (2019) to ensure stable training across noise levels. This end-to-end structure is intended This artifact-first organization also echoes resilience-oriented evaluation practice in software-intensive systems, where traceability, metric reporting, and repeatable audit structure are treated as first-class engineering requirements. To make the study straightforward to reproduce and extend (e.g., to additional noise rates or alternative schedules) without changing the core evaluation protocol**.**

4. RESULT AND DISCUSSION

Experimental Setup

We conduct experiments on Food-101 (Bossard et al., 2014), accessed through the datasets library (Lhoest et al., 2021). Food recognition is a fine-grained classification problem with substantial intra-class variation and visually similar inter-class categories, which makes it a relevant testbed for robustness under imperfect supervision. To keep experiments feasible on a single Tesla T4 GPU while still obtaining meaningful trends, we report the main robustness analysis on a 20-class subset (denoted subset20) with fixed split sizes (15,000 train / 2,500 validation / 2,500 test). This subset setting allows us to run multiple noise levels and training variants with consistent computation and to export a complete set of artifacts for auditing. In addition, we provide a short evaluation on the full 101-class setting (full101) to demonstrate that the proposed approach remains applicable beyond the reduced subset, albeit under limited training budgets.

Our backbone is EfficientNet-B0 (Tan & Le, 2019), initialized from ImageNet pretrained weights (Russakovsky et al., 2015). ImageNet pretraining is used to improve representation quality and stabilize optimization, particularly important when training labels are corrupted. All training and evaluation are implemented in the PyTorch stack (Paszke et al., 2019), with an Adam/AdamW-style optimizer (Kingma & Ba, 2015; Loshchilov & Hutter, 2019) and cosine learning-rate decay. These choices are standard for modern CNN training and help ensure that observed performance differences across methods are driven primarily by the noise-handling mechanism rather than by unstable optimization.

We emphasize strict separation of roles between data splits: validation and test sets remain clean and are used for model selection and final reporting, respectively. This design is essential because, under label corruption, training accuracy against noisy labels can become misleading and may not reflect true generalization. We therefore track clean validation metrics throughout training and use them to select the best checkpoint. Finally, in some runs, the PIL image loader reported “Truncated File Read” warnings. We observed no crashes and kept the default loader behavior; nevertheless, we report this detail for completeness because data-loading anomalies can affect reproducibility in practical pipelines.

Noise Protocol

We inject symmetric label noise (Equation 1) into training labels only, using corruption rates of 20% and 40%. Each label is independently flipped to a uniformly sampled incorrect class, producing a controlled stress test for robustness (Song et al., 2022). Importantly, both validation and test sets remain clean in all experiments. This is a critical aspect of the protocol: it preserves a trusted signal for checkpoint selection and enables valid robustness measurement under noisy supervision. In addition, noise injection is performed deterministically given a fixed seed, so that comparisons across methods at a given noise rate are not confounded by differences in which examples were corrupted.

Accuracy Under Noise (Subset20)

Table 1. Top-1 Test Accuracy on Subset20 (Clean Test). “Train Noise” Indicates the Symmetric Noise Rate Applied to Training Labels Only.

Method	Train noise	Test Acc.
Baseline (CE)	0%	0.7120
Baseline (CE)	20%	0.6476
Proposed (reweight+early)	20%	0.8176
Baseline (CE)	40%	0.5636
Proposed (reweight+early)	40%	0.6928

Table 1 reports top-1 test accuracy on the clean test set while training labels are corrupted. As expected, the baseline model trained with standard cross-entropy degrades as the noise rate increases, reflecting the increasing fraction of misleading gradient updates due to

incorrect labels. The proposed method substantially improves robustness and recovers large margins relative to the baseline, with the strongest gains observed at higher noise. This pattern is consistent with the intuition that noise-aware training should deliver increasing benefit as corruption becomes more severe, whereas standard training increasingly overfits spurious supervision.

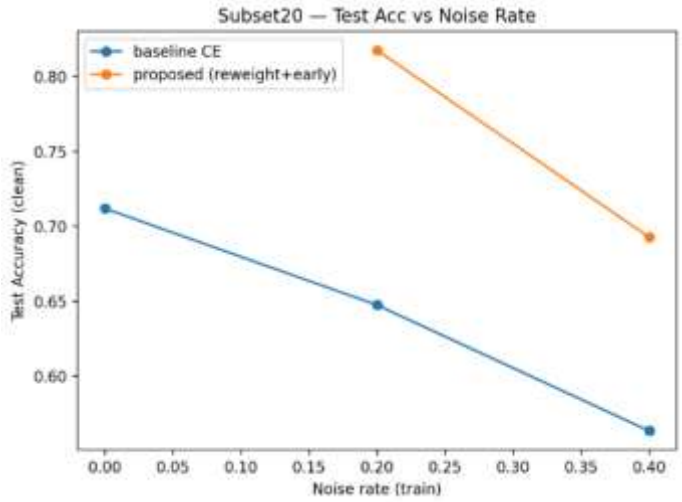

Figure 3. Accuracy vs. Noise Rate on Subset20. The Proposed Approach Maintains Higher Accuracy as Label Corruption Increases.

Figure 3 provides a compact visualization of the same trend by plotting accuracy as a function of noise rate. The figure highlights that the proposed approach maintains a higher performance regime across noise levels, suggesting that the method preserves useful learning signal even when a large fraction of the training supervision is corrupted

Training Dynamics and Memorization

While Table 1 summarizes final test outcomes, the training trajectories reveal why the proposed method helps. Deep networks often learn clean/easy patterns early and later drift into memo rization of noisy labels (Arpit et al., 2017; Liu et al., 2020). Under noisy supervision, this drift can manifest as continued improvement in training accuracy (measured against corrupted labels) even when clean validation accuracy plateaus, indicating that optimization is spending capacity f itting incorrect supervision rather than improving generalization.

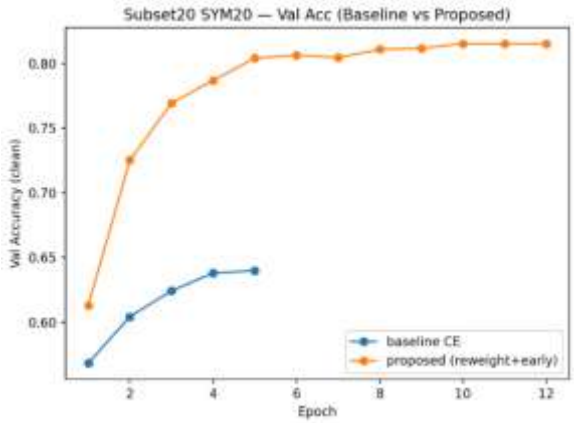

Figure 4. Validation Accuracy (Clean) Under 20% Train Noise (Subset20).

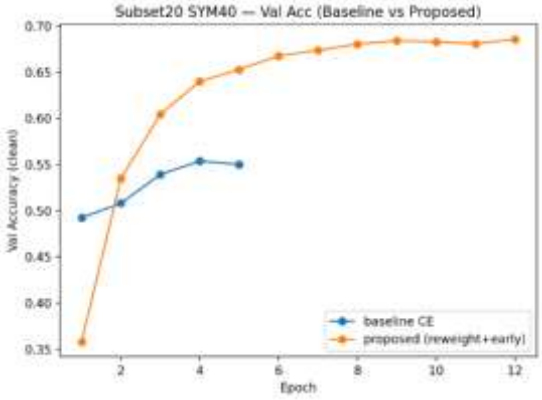

Figure 5. Validation Accuracy (Clean) Under 40% Train Noise (Subset20).

Figure 4 and Figure 5 compare clean validation accuracy trajectories under 20% and 40% noisy training labels. The baseline typically shows limited sustained improvement as noise increases, whereas the proposed method tends to achieve higher validation accuracy and to maintain a more stable trajectory, consistent with selectively emphasizing lower-loss samples and constraining late-stage memorization effects.

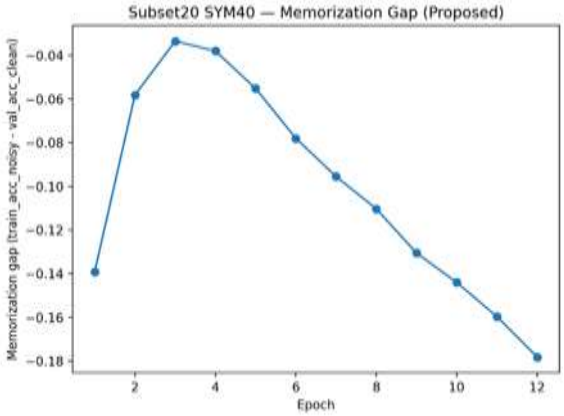

Figure 6. Memorization gap under 40% train noise (subset20, proposed). Increasing gap indicates memorization of noisy labels; the stopping rule prevents late-stage overfitting.

Figure 6 visualizes memorization behavior through the gap in Equation 4. A growing gap indicates that the model is increasingly fitting the noisy training labels without corresponding validation gains. The early-learning stopping rule is explicitly designed to detect this regime and halt training before late-stage overfitting dominates (Arpit et al., 2017; Liu et al., 2020). The trajectory therefore provides an interpretable diagnostic that links the proposed stopping criterion to observed training behavior, rather than treating early stopping as an arbitrary heuristic.

Calibration

Accuracy alone does not guarantee that predicted probabilities are trustworthy. Modern neural networks can be substantially miscalibrated, often producing overconfident predictions (Guo et al., 2017). This issue is particularly relevant under label noise: corrupted supervision can encourage the model to become confident about spurious patterns that align with noisy labels, thereby degrading reliability even when accuracy appears acceptable.

Table 2. ECE (15 bins) on subset20 (clean test). Lower is better.

Setting	Baseline (CE)	Proposed (reweight+early)
Symmetric 20%	0.105382	0.091770
Symmetric 40%	0.147401	0.081348

We evaluate calibration using reliability diagrams and Expected Calibration Error (ECE) with 15 bins (Guo et al., 2017; Naeini et al., 2015). Table 2 shows that the proposed method improves ECE, with a notably larger improvement at 40% noise. This pattern suggests that suppressing the influence of likely-noisy samples and controlling late-stage memorization not only improves classification correctness but also yields probabilities that better reflect empirical correctness likelihood. Figure 7 and Figure 8 further illustrate these trends by visualizing deviations from the diagonal (perfect calibration) across confidence bins.

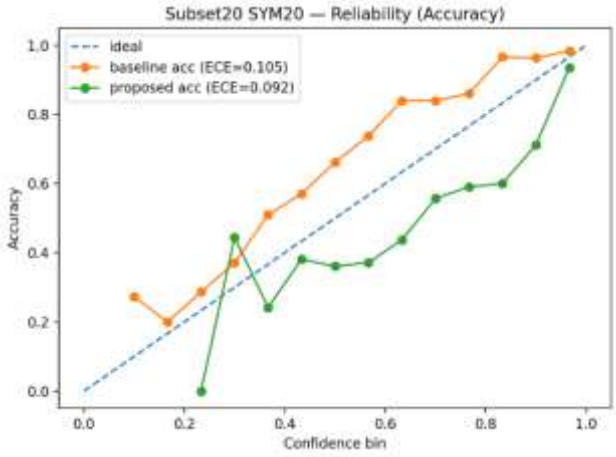

Figure 7. Reliability diagram under 20% train noise (subset20).

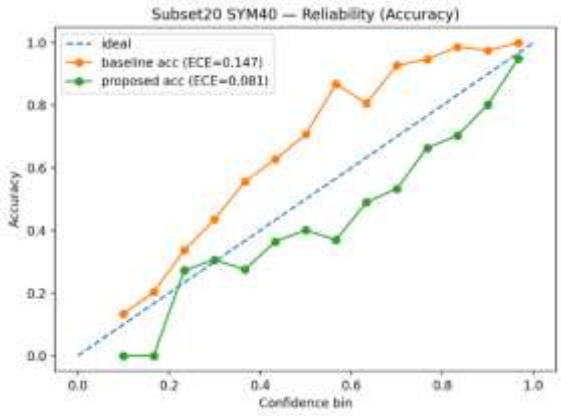

Figure 8. Reliability diagram under 40% train noise (subset20).

Runtime

Table 3. Forward-only Runtime (ms/image) on Subset20.

Setting	Tesla T4 (GPU)	CPU (tiny benchmark)
Baseline, 20% noise	5.70	81.03
Proposed, 20% noise	5.69	78.52
Baseline, 40% noise	5.55	80.21
Proposed, 40% noise	7.49	74.24

We report forward-only runtime (ms/image) to isolate inference cost from training-time overhead. Because the proposed approach changes only training-time behavior (selection/reweighting and stopping), it should add negligible inference cost relative to the baseline once a checkpoint is f ixed. Table 3 confirms that forward-only latency on GPU is broadly similar across settings, indicating that robustness gains do not require more expensive inference. This is an important practical consideration: many noisy-label methods increase training complexity (e.g., multiple networks), but what matters for deployment is often inference cost, which remains essentially unchanged here.

We additionally include a small CPU benchmark to provide a rough sense of behavior outside the GPU environment, acknowledging that absolute numbers depend on implementation, hardware, and batching. The key takeaway is qualitative: the proposed method does not fundamentally alter the inference graph and therefore preserves the efficiency advantages of EfficientNet-B0 (Tan & Le, 2019).

Qualitative Analysis

Quantitative metrics summarize performance, but qualitative inspection helps interpret where robustness gains come from and whether the method is plausibly addressing label corruption rather than merely shifting errors. We therefore provide two sets of diagnostic visualizations.

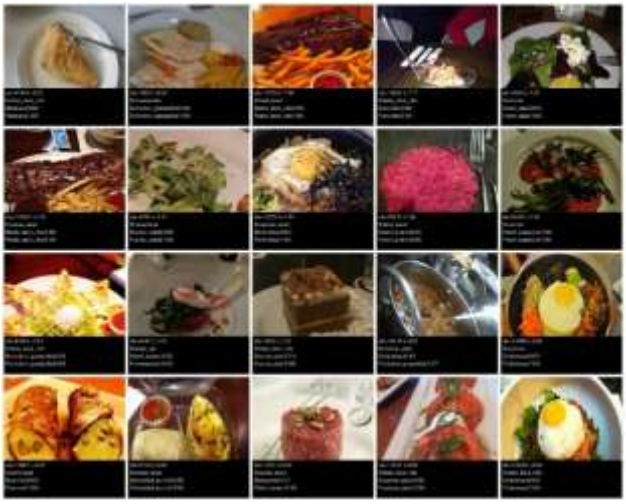

Figure 9. Top-20 high-loss training samples under 40% train noise (subset20). High loss is a strong indicator of potential label corruption; the model predictions are provided in the accompanying CSV artifact.

First, we visualize the top-20 high-loss training samples under 40% noise as noisy-label candidates (Figure 9). Under symmetric corruption, high loss is a reasonable indicator of inconsistency between the image content and the assigned (possibly flipped) label, though it can also capture genuinely hard examples. The accompanying CSV artifact includes the model predictions and metadata to support auditing and follow-up inspection.

Figure 10. Ten test failure cases (subset20): baseline vs. proposed predictions. The proposed method reduces some confusions but hard visually-similar classes remain challenging.

Second, we visualize ten representative test failures, comparing baseline and proposed predictions side-by-side (Figure 10). This comparison highlights error modes that persist despite robustness mechanisms, such as visually similar categories or ambiguous presentations, and it also shows cases where the proposed method corrects baseline confusions. These diagnostics are intended to make the empirical results more interpretable and to support informed discussion about remaining limitations.

Full 101-Class Results (Partial)

To demonstrate scalability beyond the subset setting, we also run a short experiment on the full 101-class Food-101 setup. This experiment is intentionally limited in training epochs due to compute constraints; therefore, we do not interpret the absolute numbers as a saturated performance estimate, but rather as an indication of whether the proposed mechanism transfers qualitatively to the full label space.

Table 4. Top-1 test accuracy on full101 (clean test).

Method	Train noise	Test Acc.
Baseline (CE)	0%	0.622178
Baseline (CE)	40%	0.557703
Proposed (reweight+early)	40%	0.583129

Table 4 shows that, at 40% training noise, the proposed approach still improves test accuracy relative to the baseline, although margins are smaller than in subset20. A plausible explanation is that the full101 setting is harder and the run budget is shorter, leaving less room for the selection-and-stopping mechanism to realize its full benefit. Figure 11 visualizes the same comparison between 0% and 40% noise for full101, reinforcing the conclusion that the method remains beneficial under substantial corruption even when the training budget is constrained.

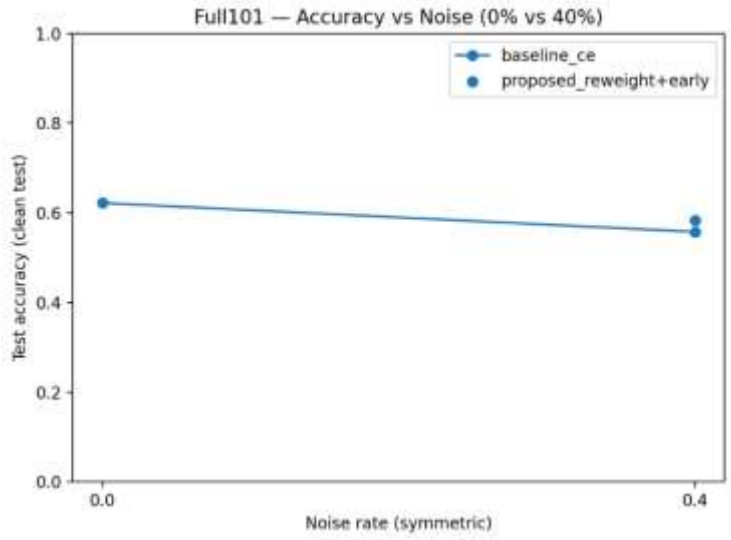

Figure 11. Full101 Accuracy at 0% vs 40% Train Noise.

Comparison

Comparison to state-of-the-art

Learning with noisy labels has evolved rapidly over the last several years, spanning a spectrum of approaches from explicit noise modeling to training-dynamics control. Early and influential directions include loss correction methods that estimate or assume a noise transition mechanism and then modify the training objective accordingly (Patrini et al., 2017). In parallel,

curriculum and selection strategies exploit the empirical observation that small-loss samples tend to be more reliable early in training, either via learned curricula (e.g., MentorNet) (Jiang et al., 2018) or via reweighting mechanisms that modulate each example's contribution to gradient updates (Ren et al., 2018). Two-network methods such as Co-teaching further address confirmation bias by having peer networks exchange selected samples (Han et al., 2018). More recently, semi-supervised formulations like DivideMix combine mixture modeling and iterative refinement to separate clean and noisy subsets and leverage unlabeled-style objectives (Li et al., 2020). In terms of pure performance, these more complex pipelines can be very strong, but they are typically accompanied by higher compute cost, more moving parts, and greater sensitivity to hyperparameter tuning.

Relative to this landscape, our approach is intentionally conservative in ambition and aggressive in practicality. The primary design constraint is not to compete with the most elaborate pipelines, but to deliver measurable robustness and reliability improvements under a strict compute budget and a simple experimental protocol. The method therefore emphasizes mechanisms that can be implemented and audited easily: small-loss driven emphasis and an early-learning control signal motivated by memorization behavior (Arpit et al., 2017; Liu et al., 2020). This positioning leads to three concrete differences in practice: a) single-model training with minimal overhead. Unlike Co-teaching and many semi supervised pipelines (Han et al., 2018; Li et al., 2020), we train only a single EfficientNet-B0 instance. This choice avoids the additional GPU memory footprint and wall-clock cost of multi-network training and simplifies experiment management. Under a single T4 GPU constraint, this difference is not cosmetic; it determines whether multiple noise levels and ablations can be executed reliably with complete artifact export, b) no mixture modeling or iterative pseudo-labeling loops. Semi-supervised meth ods commonly rely on iterative refinement stages and distributional modeling of losses or predictions (Li et al., 2020). While effective, these pipelines often introduce additional hyperparameters (e.g., warmup length, mixture thresholds, augmentation choices) and can be sensitive to implementation details. By contrast, our approach relies on a direct and interpretable signal per-example loss ranking and a simple training-dynamics control rule, reducing implementation complexity and the surface area for hidden confounders, c) explicit memorization control grounded in early-learning behavior. The literature emphasizes that late-stage training can lead to memorization of noisy labels (Arpit et al., 2017; Liu et al., 2020). We operationalize this observation with a practical memorization-gap monitor that is easy to compute and easy to audit. This is not a new theoretical contribution; rather, it is an

implementation-level decision aimed at making early-learning regularization behavior measurable and actionable in a constrained experimental setting.

Overall, the comparison is best understood as a trade-off. Heavier methods can deliver stronger peak results when compute and engineering budget are ample, whereas our method is designed to be an auditable and reproducible improvement that remains feasible under limited hardware. In that sense, our contribution is practical: demonstrating that meaningful robustness and calibration gains are achievable without adopting complex multi-stage pipelines.

Measured Contribution

Empirically, our method delivers large gains in the regime where noisy supervision is most damaging. On subset20 with 40% symmetric noise, the proposed approach improves top-1 test accuracy from 0.5636 (baseline CE) to 0.6928, a substantial recovery in correctness under heavy label corruption. In the same setting, calibration improves markedly: ECE decreases from 0.1474 to 0.0813, indicating that predicted probabilities are substantially more aligned with empirical accuracy (Guo et al., 2017; Naeini et al., 2015). These results are consistent with the method's design intent: suppress gradients from likely-noisy samples and stop training before late-stage memorization dominates (Arpit et al., 2017; Liu et al., 2020).

Equally important from a deployment perspective, the gains do not require a more expensive inference model. The selection and stopping logic operates during training only; inference uses the same EfficientNet-B0 architecture and therefore retains the same forward-pass cost. This matters because many robustness methods increase training complexity and sometimes also inference overhead if ensembling or auxiliary modules are introduced. Here, robustness is obtained primarily through training-time control of which examples influence the model and when optimization should stop, rather than by changing the model class itself.

Limitations

This study has clear limitations, and they are largely driven by design constraints rather than by oversight. First, we do not provide a direct numerical leaderboard-style comparison against heavy pipelines such as DivideMix or two-network Co-teaching on Food-101 (Han et al., 2018; Li et al., 2020). The reason is pragmatic: our experimental goal is a single-model, compute-feasible pipeline that can be executed on a single T4 GPU with complete artifact export and auditable reproducibility. Re-implementing or faithfully reproducing complex multi-stage pipelines under the same constraints would either exceed compute budgets or compromise the controlled nature of the benchmark through additional tuning degrees of freedom.

Second, the robustness conclusions are established primarily under symmetric synthetic noise. This is a standardized stress test and is useful for reproducibility, but it does not capture all real-world noise structures (e.g., class-dependent confusions). Extending the evaluation to more realistic noise models and larger training budgets would strengthen the external validity of the f indings.

Third, while the proposed method is motivated by early-learning and memorization behavior (Arpit et al., 2017; Liu et al., 2020), it remains a heuristic training recipe rather than a theoretically optimal solution. Its effectiveness depends on the reliability of the small-loss signal and on having a clean validation set for checkpoint selection. In future work, the approach could be combined with stronger backbones, longer training, or integrated into peer-learning frameworks such as Co-teaching to reduce single-model selection bias (Han et al., 2018). It may also be complemented by robust-loss or correction ideas (Patrini et al., 2017; Zhang & Sabuncu, 2018) when additional modeling complexity is acceptable.

5. CONCLUSIONS

Summary of findings

This study examined food recognition under controlled synthetic label noise on Food-101 (Bossard et al., 2014) and confirmed a clear, practically important trend: a standard EfficientNet-B0 classifier trained with cross-entropy becomes increasingly brittle as the fraction of corrupted training labels grows. In the controlled benchmark where validation and test labels remain clean, we observe that increasing symmetric noise produces sharp degradation in generalization, consistent with the intuition that noisy labels inject systematically misleading gradients that standard empirical risk minimization cannot reliably ignore.

To address this, we proposed a lightweight training recipe that targets noisy-label failure modes through training-time control rather than architectural changes. The approach combines (i) small-loss driven sample emphasis, leveraging the empirical tendency that clean/easy samples yield smaller losses early in training, and (ii) an early-learning stopping criterion based on a memorization-gap signal, motivated by evidence that deep networks can transition from learning true structure to memorizing noise in later epochs (Arpit et al., 2017; Liu et al., 2020). Across symmetric noise rates of 20% and 40%, the proposed method improves top-1 test accuracy relative to the baseline and yields more reliable probabilities, reflected in improved calibration metrics. Importantly, these gains do not require any increase in inference

complexity: the final model is still EfficientNet-B0 (Tan & Le, 2019), and the robustness mechanism operates during training only.

Beyond final metrics, the training dynamics provide an interpretable explanation for the observed improvements. Under label corruption, training accuracy against noisy labels can continue rising even after validation accuracy saturates, suggesting increasing memorization rather than improved generalization (Arpit et al., 2017). The memorization-gap monitor operationalizes this behavior into a practical stopping decision, and the empirical trajectories align with the intended effect: halting training near the onset of late-stage noise fitting while preserving early generalization gains.

Implications

The results support two practical implications for learning under noisy supervision. First, per-sample loss can serve as a useful and implementable proxy for label reliability during the early learning phase. This does not imply that low-loss samples are always clean or that high-loss samples are always mislabeled; rather, the loss ranking provides a training signal that can be exploited to reduce the average harm of corrupted supervision, especially when the noise rate is substantial. This perspective is consistent with curriculum and selection ideas used in noisy-label learning and highlights that robust behavior can sometimes be obtained through simple training-time control without complex noise modeling pipelines.

Second, explicit early-learning control appears to be an effective mechanism to mitigate late-stage memorization of corrupted labels. Early-learning regularization emphasizes that late optimization can be actively harmful under noise (Liu et al., 2020), and our memorization-gap criterion provides a concrete, auditable operationalization of this principle. From an engineering standpoint, this matters because many state-of-the-art noisy-label approaches rely on multiple networks, iterative semi-supervised refinement, or mixture modeling (Han et al., 2018; Li et al., 2020), which may be infeasible in resource-constrained environments. A single-model approach that delivers measurable robustness and calibration gains is therefore attractive for deployments where GPU memory, training time, and implementation complexity are primary constraints.

Limitations and Future Work

This study has several limitations that point to clear next steps. First, we do not include a dedicated ablation that isolates the effect of small-loss selection/reweighting from the effect of the early-learning stopping rule. While the combined method improves performance, a clean ablation (e.g., reweight-only vs. stopping-only vs. combined) would quantify which component

contributes most and under what noise regimes. Such decomposition would strengthen causal claims about the mechanism.

Second, our experiments primarily focus on symmetric synthetic noise, which is a standardized and reproducible stress test but does not capture all real-world noise structures. In practice, label corruption is often class-dependent (systematic confusions) or instance-dependent (hard examples mislabeled more frequently). Extending the evaluation to class-dependent noise and broader collection biases would improve external validity and clarify how well the method generalizes beyond uniform corruption.

Third, the study emphasizes a compute-feasible single-model pipeline and therefore does not provide a full leaderboard-style comparison against heavier state-of-the-art pipelines such as Co-teaching or DivideMix (Han et al., 2018; Li et al., 2020), nor does it exhaustively tune longer full101 runs. Future work should evaluate performance ceilings under larger training budgets and include stronger baselines when compute and engineering resources permit. In addition, robustness evaluation could be complemented with explicit input-corruption benchmarks to probe generalization under distribution shift (Hendrycks & Dietterich, 2019). Finally, abstention-based handling can be a practical reliability enhancement in open-world settings, enabling the system to defer on uncertain samples rather than forcing an overconfident prediction (Thulasidasan et al., 2019). Combining abstention policies with improved calibration and noisy-label robustness is a promising direction for building more trustworthy food recognition systems.

REFERENCES

Arazo, E., Ortenzi, F., Albert, P., O'Connor, N. E., & McGuinness, K. (2019). *Unsupervised label noise modeling and loss correction*. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Arpit, D., Jastrzebski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., et al. (2017). *A closer look at memorization in deep networks*. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Bossard, L., Guillaumin, M., & Van Gool, L. (2014). Food-101: Mining discriminative components with random forests. In *European Conference on Computer Vision (ECCV)* (pp. 446–461). https://doi.org/10.1007/978-3-319-10599-4_29

Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). *On calibration of modern neural networks*. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I. W., & Sugiyama, M. (2018). *Co-teaching: Robust training of deep neural networks with extremely noisy labels*. In *Advances in Neural Information Processing Systems (NeurIPS)*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/CVPR.2016.90

Hendrycks, D., & Dietterich, T. (2019). *Benchmarking neural network robustness to common corruptions and perturbations*. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Jiang, L., Zhou, Z., Leung, T., Li, L.-J., & Fei-Fei, L. (2018). *MentorNet: Learning data-driven curriculum for very deep neural networks on corrupted labels*. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Kingma, D. P., & Ba, J. (2015). *Adam: A method for stochastic optimization*. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Kuleshov, V., Fenner, N., & Ermon, S. (2018). *Accurate uncertainties for deep learning using calibrated regression*. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Lhoest, Q., Villanova del Moral, A., Jernite, Y., Thakur, A., von Platen, P., Patil, S., et al. (2021). Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP)* (pp. 175–184). https://doi.org/10.18653/v1/2021.emnlp-demo.21

Li, J., Socher, R., & Hoi, S. C. H. (2020). *DivideMix: Learning with noisy labels as semi-supervised learning*. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Liu, S., Niu, G., & Sugiyama, M. (2020). *Early-learning regularization prevents memorization of noisy labels*. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Loshchilov, I., & Hutter, F. (2019). *Decoupled weight decay regularization*. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Müller, R., Kornblith, S., & Hinton, G. (2019). *When does label smoothing help?* In *Advances in Neural Information Processing Systems (NeurIPS)*.

Naeini, M. P., Cooper, G., & Hauskrecht, M. (2015). *Obtaining well-calibrated probabilities using Bayesian binning*. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). *PyTorch: An imperative style, high-performance deep learning library*. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Patrini, G., Rozza, A., Menon, A. K., Nock, R., & Qu, L. (2017). Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1944–1952). https://doi.org/10.1109/CVPR.2017.240

Ren, M., Zeng, W., Yang, B., & Urtasun, R. (2018). *Learning to reweight examples for robust deep learning*. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision, 115*(3), 211–252. https://doi.org/10.1007/s11263-015-0816-y

Song, H., Kim, M., Park, D., Shin, Y., Lee, J.-G., et al. (2022). Learning from noisy labels: A survey. *IEEE Transactions on Neural Networks and Learning Systems*.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/CVPR.2016.308

Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 6105–6114).

Thulasidasan, S., Bhattacharya, T., Bilmes, J. A., Chennupati, G., & Mohd-Yusof, J. (2019). *Combating label noise in deep learning using abstention*. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Zhang, Z., & Sabuncu, M. R. (2018). *Generalized cross entropy loss for training deep neural networks with noisy labels*. In *Advances in Neural Information Processing Systems (NeurIPS)*.