An Empirical Study of Normalized Loss Functions and Active-Passive Learning for Training CNNs with Noisy Labels on CIFAR-10

Aditya Nagarsekar

Abstract

Training Deep Neural Networks (DNNs) effectively is often hampered by the presence of noisy (incorrect) labels in datasets, a common issue in real-world scenarios. The standard Cross Entropy (CE) loss function is known to be sensitive to such noise, potentially leading to degraded model performance by overfitting incorrect labels. Inspired by recent theoretical advancements [6], this paper investigates strategies to mitigate the impact of noisy labels. We empirically evaluate two core ideas: (1) the principle that normalizing any loss function can induce robustness against label noise, and (2) the Active Passive Loss (APL) framework, which combines a robust active loss (focusing on the labeled class) with a robust passive loss (considering other classes). This combination is specifically designed to overcome the underfitting issues observed in some purely robust or normalized losses, thereby achieving both noise tolerance and sufficient learning capacity. This study implements and evaluates these concepts using a standard Convolutional Neural Network (CNN) architecture on the CIFAR-10 benchmark dataset under controlled levels of both symmetric (random errors) and asymmetric (class confusion) label noise. We compare standard losses (CE, Focal Loss), their normalized counterparts (NCE, NFL), baseline robust losses (MAE, RCE) and their normalized versions (NMAE, NRCE), and various APL combinations (e.g., NCE+MAE, NCE+RCE). Our findings indicate that while normalization effectively imparts robustness, APL methods, particularly NCE+MAE and NFL+RCE under symmetric noise, and NCE+RCE and NFL+RCE under asymmetric noise, demonstrate significantly improved performance. They achieve a superior balance between noise tolerance and learning capacity, especially under high noise rates. This provides strong empirical support for the APL framework as a principled and effective approach.

1 Introduction

Deep Neural Networks (DNNs) have achieved remarkable success across various domains, yet their performance heavily relies on the availability of large datasets with accurately labeled data. In practice, obtaining perfectly clean labels is often expensive or infeasible, leading to datasets contaminated with noisy labels [4]. Training DNNs directly on such noisy data can cause the model to memorize the incorrect labels, leading to overfitting and poor generalization performance on clean, unseen data.

The standard loss function for classification, Cross Entropy (CE), while effective on clean datasets due to its alignment with maximum likelihood estimation, is particularly susceptible to noisy labels. It strongly encourages the model to assign high confidence to the given (potentially incorrect) label, thereby propagating the label errors into the model's parameters [1]. This inherent limitation has spurred research into alternative approaches for robust learning in the presence of label noise.

Recent theoretical work by Ma et al. [6] introduced compelling ideas for designing robust loss functions, forming the primary inspiration for this study. Their first key contribution was demonstrating that a simple, theoretically grounded normalization procedure (see Section 3.4) can make *any* loss function robust to label noise under certain conditions (specifically, noise rates below a certain threshold). This offers a potentially universal mechanism for adapting existing, well-understood losses (like CE or FL [5]) for noisy settings. However, they also insightfully noted that merely achieving robustness might not be sufficient for optimal performance. Some robust losses, including potentially the normalized ones, can suffer from an underfitting problem. This manifests as slow convergence or the model failing to learn the underlying data distribution effectively, potentially because the robust loss provides weaker or less discriminative gradients compared to non-robust losses on clean data. To address this crucial trade-off between robustness and learning capacity, Ma et al. proposed the Active Passive Loss (APL) framework. APL combines two complementary robust loss functions: an active loss that primarily maximizes the probability of the given label (pushing the correct signal), and a passive loss that explicitly minimizes probabilities of other classes (suppressing noise and enforcing discrimination). By using robust versions of both active and passive components, the APL framework aims to achieve a synergy that yields both strong noise tolerance and a sufficient learning signal to avoid underfitting.

This paper presents an empirical study focused on validating and comparing these concepts within a controlled experimental setting. Our specific contributions are:

- We empirically verify the robustness induced by normalizing standard losses like CE and Focal Loss (FL), resulting in Normalized Cross Entropy (NCE) and Normalized Focal Loss (NFL).
- We investigate the learning dynamics and potential underfitting issues associated with baseline robust losses (Mean Absolute Error (MAE) [1] and Reverse Cross Entropy (RCE) [10]) and the normalized losses (NCE, NFL, NMAE, NRCE).
- We implement and evaluate the APL framework by combining normalized active losses (NCE, NFL) with robust or normalized passive losses (MAE, RCE), creating specific combinations like NCE+MAE, NCE+RCE, NFL+MAE, and NFL+RCE.
- We conduct extensive experiments on the CIFAR-10 benchmark [4] using a standard CNN architecture, under controlled symmetric (0% to 80%) and asymmetric (0% to 40%) noise models.

Our results, consistent with [6], provide practical insights into the effectiveness and trade-offs of normalization and the APL framework, guiding the choice of loss functions when training DNNs on noisy datasets.

2 Related Work

Learning with noisy labels is addressed via strategies including [6]:
1) Label Correction [9],
2) Loss Correction [7],
3) Robust Training Strategies [2, 3], and
4) Robust Loss Functions (our focus).

MAE [1] and RCE [10] are baseline robust passive losses. Related work includes GCE (Generalised Cross Entropy) [11] and SCE (Symmetric Cross Entropy) [10].
Our study tests Ma et al.'s specific framework [6]: systematic normalization (Eq. 1) and APL using *robust* components (e.g., NCE/NFL + MAE/RCE).

3 Methodology

This section details the experimental framework, including the dataset, noise models, network architecture, loss functions implemented, evaluation setup, and the Active Passive Loss (APL) framework.

3.1 Preliminaries

We address $K = 10$ class classification on the CIFAR-10 dataset [4]. For an input image x, the model outputs logits $f_\theta(x)$, which are typically transformed into class probabilities $p(k|x)$ for $k = 1...K$ using the softmax function. The provided training label for x is denoted by y (the class index), which may be noisy, with $q(x)$ representing its one-hot vector form. The ground truth (clean) label is assumed unknown during training.

We investigate two standard noise models to simulate different real-world scenarios:

- **Symmetric Noise:** Models random annotation errors. Each label has a probability η of being flipped, and if flipped, it changes to any of the $K - 1$ incorrect classes with equal probability $\eta/(K - 1)$.

- **Asymmetric Noise:** Models systematic confusion between visually similar classes. Following common practice [7, 10], specific class pairs (e.g., TRUCK → AUTOMOBILE, BIRD → AIRPLANE, DEER → HORSE, CAT ↔ DOG) are flipped with probability η. See Appendix A.2 for the full mapping.

3.2 Network Architecture

To evaluate the different loss functions, we employ a standard 8-layer Convolutional Neural Network (CNN) architecture inspired by the VGG pattern [8], adapted for use on the CIFAR-10 dataset [4]. The detailed structure, named `CNN_Model` in our implementation, is visualized in Appendix A.1. This architecture was chosen because:

- It represents a common and reasonably effective baseline for image classification tasks on CIFAR-10.
- Its complexity is sufficient to demonstrate the effects of different loss functions under noisy labels without being overly large or computationally prohibitive.
- Using a standard architecture allows the focus of the study to remain clearly on the impact and comparison of the loss functions themselves, rather than on novel architectural contributions.

The network consists of stacked convolutional layers with 3x3 kernels, Batch Normalization, ReLU activations, followed by Max Pooling layers, and concludes with a classifier head composed of fully connected layers.

3.3 Implemented Base Loss Functions

We evaluate a selection of representative loss functions, categorized based on the APL framework [6] as either primarily active (focusing on the target class probability $p(y|x)$) or passive (considering other classes or the relationship $1 - p(y|x)$). **Unless otherwise specified, the logarithm** ($\log$) **used in these functions refers to the natural logarithm (base** e**).**

Active Losses: Chosen for their widespread use and known properties.

- **Cross Entropy (CE)**: The standard baseline for classification, known for its effectiveness on clean data but sensitivity to noise. $L_{\text{CE}} = -\log(p(y|x))$.
- **Focal Loss (FL)** [5]: An adaptation of CE designed to down-weight easily classified examples, often used for class imbalance. Its interaction with noisy labels is also of interest. We use $\gamma = 0.5$. $L_{\text{FL}} = -(1 - p(y|x))^\gamma \log(p(y|x))$.

Passive Losses: Chosen as established examples of robust loss functions.

- **Mean Absolute Error (MAE)** [1]: Theoretically shown to be noise-robust under certain conditions; considered a boundary-seeking loss. We use the common form $L_{\text{MAE}} = 2(1 - p(y|x))$.
- **Reverse Cross Entropy (RCE)** [10]: Another robust loss that encourages low probabilities for non-target classes, operating differently from MAE. We use the formulation with target pseudo-labels for non-target classes, controlled by parameter $A = -4.0$. $L_{\text{RCE}} = -\sum_{k=1}^{K} p(k|x) \log(q'_k(x))$, where $q'_k(x) = \exp(A)$ if $k \neq y$, else 1.

3.4 Normalized Loss Functions

Following Ma et al. [6], we apply a normalization procedure intended to confer noise robustness to any base loss function l. The normalized loss L_{norm} is defined as:

$$L_{\text{norm}}(f_\theta(x), y) = \frac{l(f_\theta(x), y)}{\sum_{j=1}^{K} l(f_\theta(x), j)} \tag{1}$$

The intuition is that by scaling the loss contribution of the target label y relative to the sum of losses over all possible labels j, the influence of a potentially incorrect y is mitigated, especially if the model assigns high loss (low probability) to the true underlying class. Figure 1 illustrates this process.

We implemented normalized versions of all four base losses: Normalized CE (NCE), Normalized FL (NFL), Normalized MAE (NMAE), and Normalized RCE (NRCE). As detailed in Appendix A.3, NMAE and NRCE simplify under certain assumptions, whereas NCE and NFL generally do not have such simple closed forms involving only $p(y|x)$.

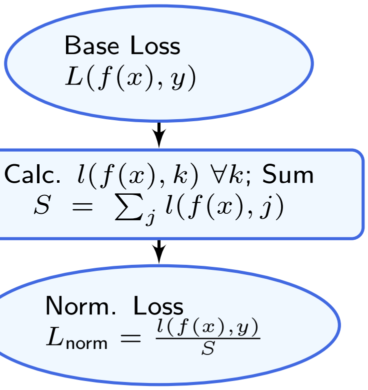

Base Loss
$L(f(x), y)$
Calc. $l(f(x), k)$ $\forall k$; Sum
$S = \sum_j l(f(x), j)$
Norm. Loss
$L_{\text{norm}} = \frac{l(f(x),y)}{S}$

Figure 1: Loss normalization process based on Eq. 1.

3.5 Active Passive Loss (APL) Framework

While normalization can provide robustness, it may sometimes lead to underfitting (slow convergence or suboptimal final accuracy). The APL framework [6] aims to overcome this by combining a robust active loss with a robust passive loss:

$$L_{\text{APL}} = \alpha L_{\text{Active}} + \beta L_{\text{Passive}} \tag{2}$$

The synergy arises because the active term provides a primary learning signal (albeit potentially noisy), while the passive term enhances discrimination against non-target classes and further mitigates the noise sensitivity, theoretically allowing for both robustness and sufficient learning capacity. Figure 2 illustrates this combination.

Crucially, the APL framework advocates using *robust* versions for both components. We test combinations using normalized active losses (NCE, NFL) and robust passive losses (MAE, RCE): NCE+MAE, NCE+RCE, NFL+MAE, and NFL+RCE. Following the baseline setup in [6], we use equal weighting $\alpha = 1.0, \beta = 1.0$ for evaluation, acknowledging that tuning these hyperparameters could potentially yield further improvements.

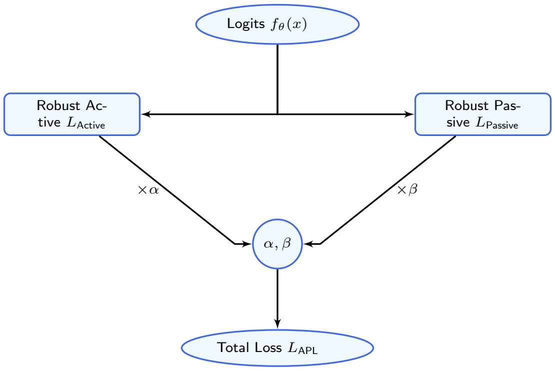

Logits $f_\theta(x)$
Robust Active L_{Active}
Robust Passive L_{Passive}
$\times\alpha$
$\times\beta$
α, β
Total Loss L_{APL}

Figure 2: APL framework concept (Eq. 2).

4 Experiments

This section details the experimental setup and presents the results for both symmetric and asymmetric noise scenarios on the CIFAR-10 dataset.

4.1 Setup

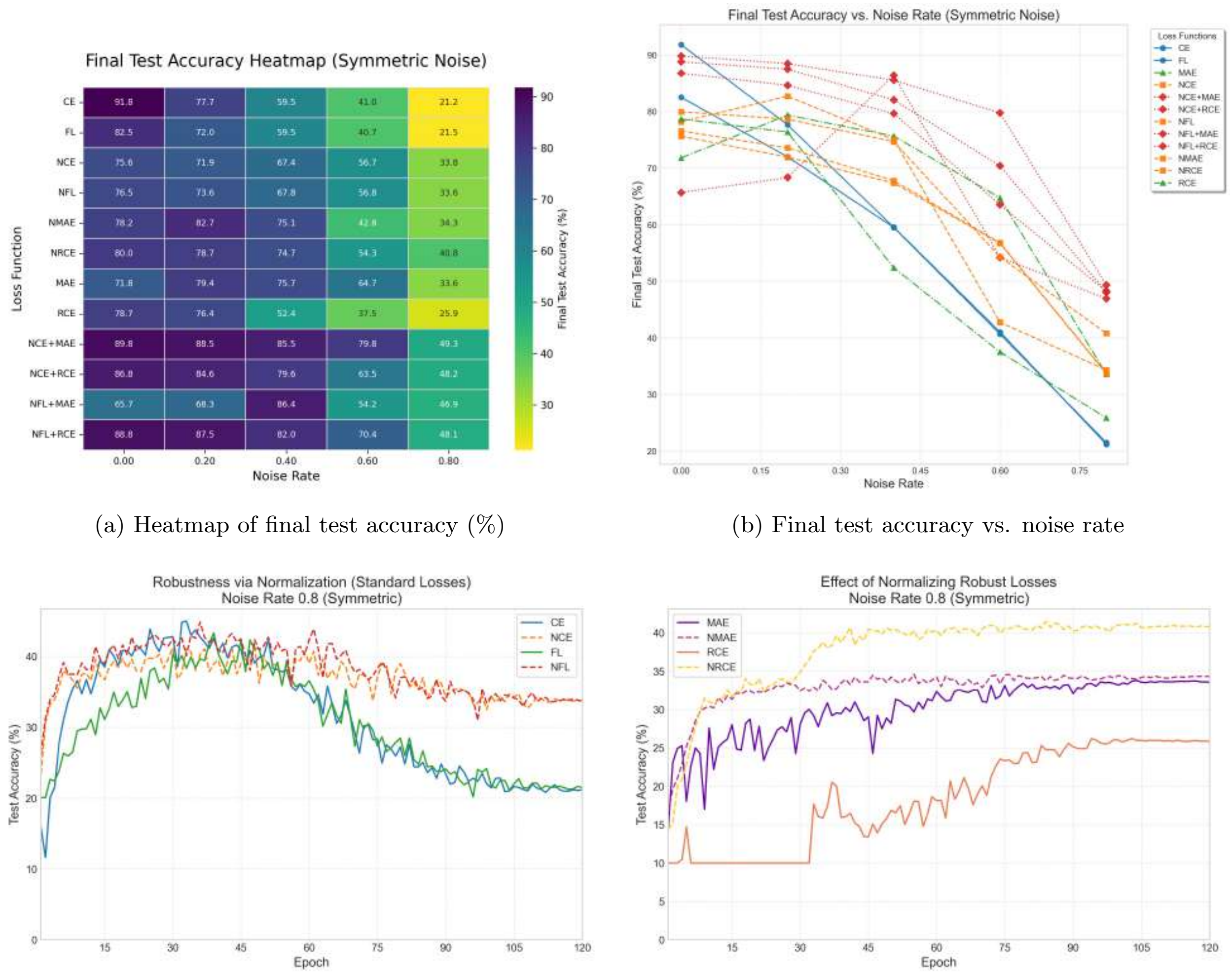
Dataset: CIFAR-10 [4], using the standard 50k training / 10k test split.
Noise Generation: Symmetric ($\eta \in \{0.0, 0.2, 0.4, 0.6, 0.8\}$) and Asymmetric ($\eta \in \{0.0, 0.1, 0.2, 0.3, 0.4\}$) noise added to training labels via `CIFAR10Noisy` (details in Appendix A.2).
Test set clean.
Network Architecture: `CNN_Model` (Appendix A.1).
Training: 120 epochs, SGD (momentum=0.9, decay=10^{-4}), LR=0.01 + Cosine Anneal, Batch=128. AMP enabled for all symmetric noise runs (except 0.8 noise NCE + RCE), disabled for all asymmetric noise runs.
Loss Params: $\gamma = 0.5$ (FL/NFL), $A = -4.0$ (RCE/NRCE), $\alpha = 1.0, \beta = 1.0$ (APL).
Metric: Final test accuracy (%).

4.2 Results: Symmetric Noise

Table 1 shows final test accuracies. Figures 3 and 4 visualize overall trends, normalization effects, overfitting, and stability. Figure 5 compares APL combinations against their components.

Table 1: Symmetric Noise: Final Test Accuracies (%). Bold = Top 2 per noise rate.

Loss Function	Clean (0.0)	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.6$	$\eta = 0.8$
CE	91.80	77.66	59.54	41.03	21.19
FL (γ=0.5)	82.52	72.01	59.52	40.66	21.50
MAE	71.81	79.37	75.66	64.74	33.58
RCE (A=-4)	78.68	76.39	52.40	37.51	25.85
NCE	75.64	71.94	67.37	56.71	33.77
NFL (γ=0.5)	76.55	73.58	67.78	56.80	33.65
NMAE	78.20	82.70	75.06	42.75	34.33
NRCE (A=-4)	79.95	78.66	74.68	54.27	40.82
NCE+MAE (APL)	89.83	**88.46**	**85.53**	**79.75**	**49.30**
NCE+RCE (APL)	86.78	84.63	79.64	63.51	48.23
NFL+MAE (APL)	65.68	68.34	**86.41**	54.22	46.94
NFL+RCE (APL)	88.78	**87.49**	82.05	**70.40**	**48.08**



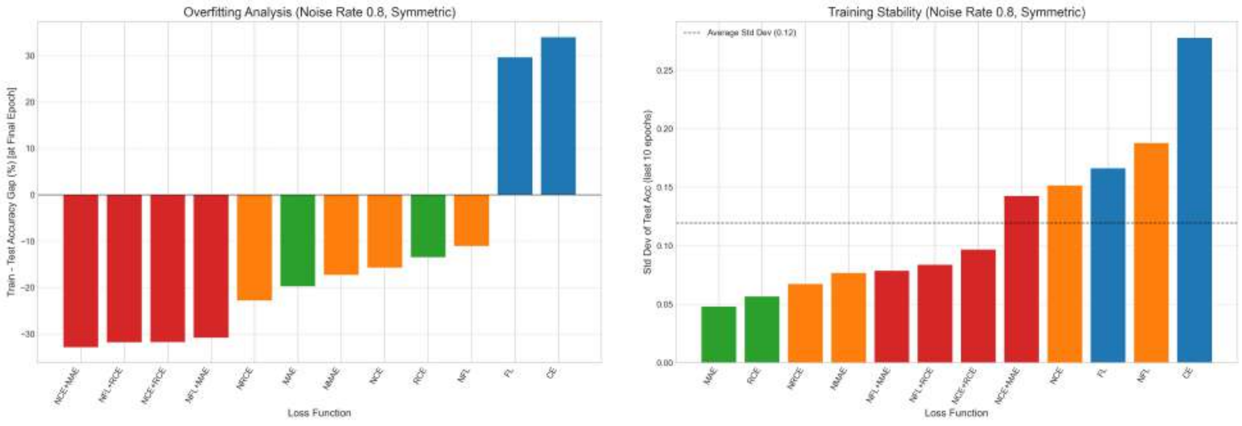
(a) Heatmap of final test accuracy (%)

(b) Final test accuracy vs. noise rate

(c) Normalization Effect (CE/FL, $\eta = 0.8$)

(d) Normalization Effect (MAE/RCE, $\eta = 0.8$)

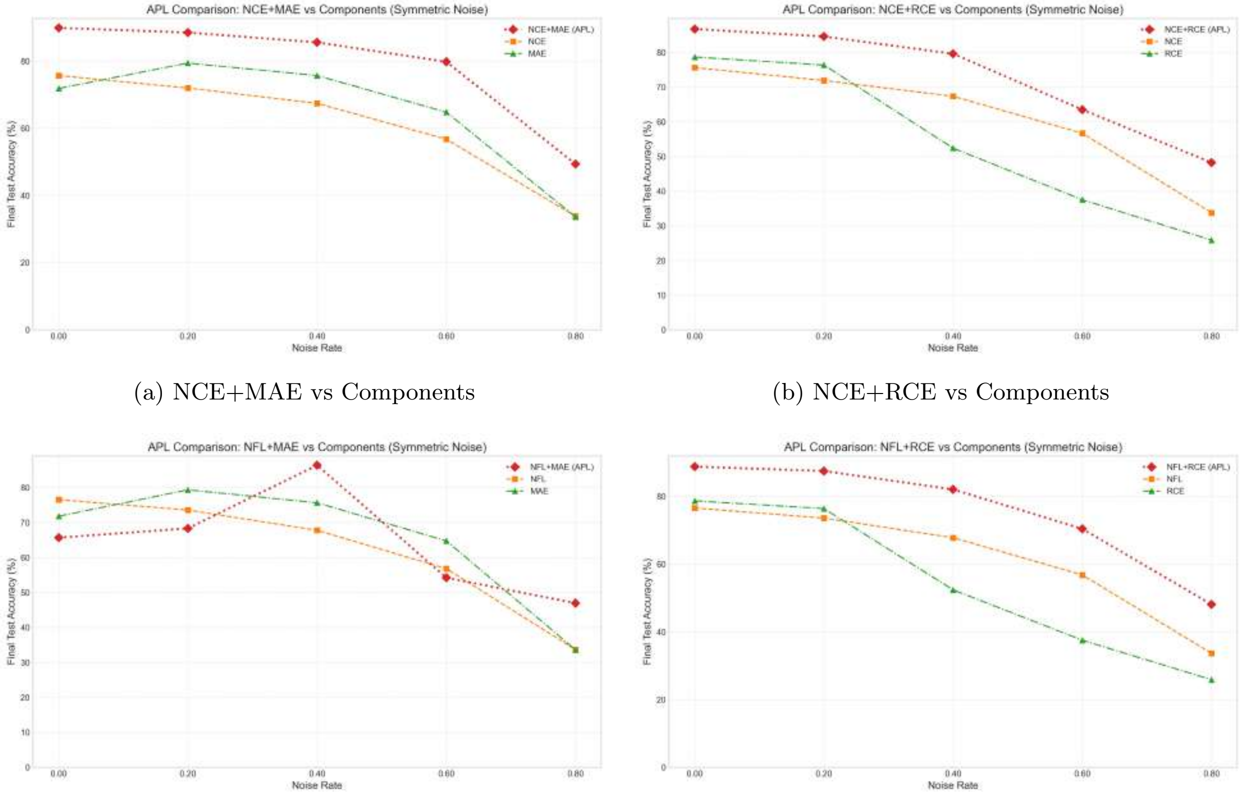
Figure 3: Symmetric noise performance visualizations. (a) Heatmap summarizes performance. (b) Line plot shows accuracy vs. noise. (c, d) Learning curves illustrate normalization impact at high noise ($\eta = 0.8$).



(a) Overfitting Gap (Train-Test Acc, $\eta = 0.8$)

(b) Training Stability ($\eta = 0.8$)

Figure 4: Symmetric noise analysis at $\eta = 0.8$. (a) Overfitting gap. (b) Training stability (Std Dev of Test Acc over last 10 epochs).



(a) NCE+MAE vs Components

(b) NCE+RCE vs Components

(c) NFL+MAE vs Components

(d) NFL+RCE vs Components

Figure 5: APL synergy under symmetric noise. APL combinations (red diamonds) are compared with their constituent active (orange squares) and passive (green triangles) losses across noise rates.

Observations (Symmetric): Our results largely corroborate the findings of Ma et al. [6].

- **Standard Losses Overfit:** CE and FL show strong performance on clean data but degrade severely with increasing noise, exhibiting significant overfitting (see large positive gap in Fig. 4a).

- **Normalization Imparts Robustness but Underfits:** NCE and NFL are much more robust to noise, preventing the accuracy collapse seen in CE/FL (Fig. 3c). However, their final accuracy often plateaus below the best APL methods, especially at higher noise rates, suggesting underfitting. Normalizing MAE/RCE did not consistently improve performance and sometimes worsened underfitting (Fig. 3d, 3b), possibly due to the simple scaling or inherent limitations.
- **APL Excels:** APL combinations, particularly NCE+MAE and NFL+RCE, achieve the best or near-best performance across most noise levels. NCE+MAE and NFL+RCE are exceptionally strong at high noise rates ($\eta = 0.6, 0.8$), while NCE+RCE performs well up to $\eta = 0.6$. This highlights the APL framework's ability to leverage the strengths of both components (Fig. 5).
- **Stability:** APL methods, along with MAE and RCE, show relatively good training stability (low std dev in late epochs) compared to the overfitting methods CE/FL/NCE/NFL (Fig. 4b).

Detailed learning curves are in Appendix A.4.

4.3 Results: Asymmetric Noise

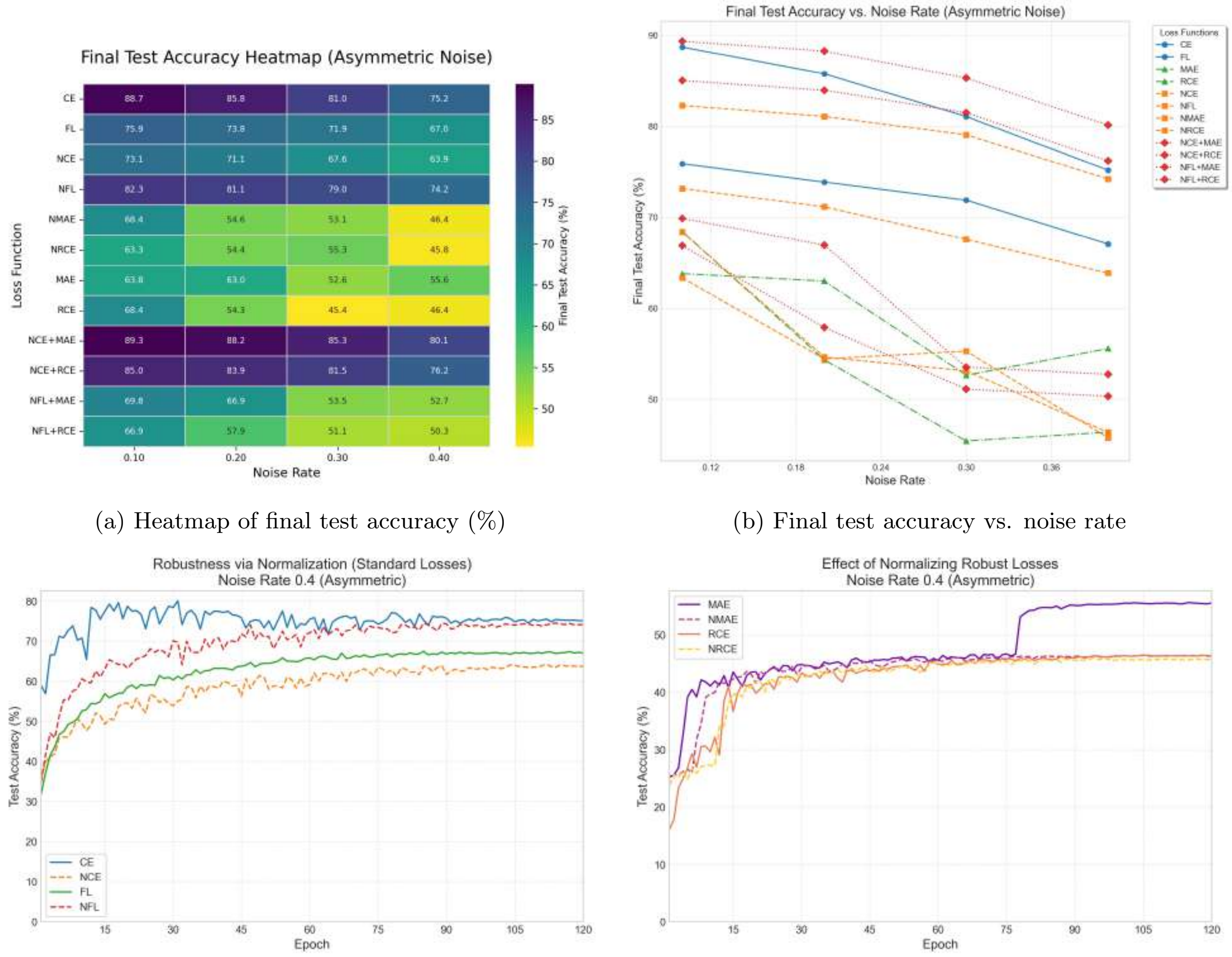
Table 2 presents the final test accuracies under asymmetric noise. Figure 6 and 7 provide visual summaries and analysis.

Table 2: Asymmetric Noise: Final Test Accuracies (%). Bold = Top 2 per noise rate.

Loss Function	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.3$	$\eta = 0.4$
CE	**88.67**	**85.75**	81.05	75.15
FL (γ=0.5)	75.86	73.85	71.86	67.05
MAE	63.79	62.98	52.64	55.56
RCE (A=-4)	68.42	54.33	45.41	46.39
NCE	73.14	71.13	67.56	63.85
NFL (γ=0.5)	82.27	81.06	79.04	74.19
NMAE	68.36	54.63	53.11	46.36
NRCE (A=-4)	63.33	54.43	55.28	45.76
NCE+MAE (APL)	**89.31**	**88.24**	**85.28**	**80.10**
NCE+RCE (APL)	85.01	83.94	**81.47**	**76.17**
NFL+MAE (APL)	69.85	66.93	53.52	52.73
NFL+RCE (APL)	66.87	57.89	51.10	50.33

Observations (Asymmetric):

- Asymmetric noise generally degrades performance less severely than symmetric noise for standard CE/FL at equivalent η values (up to 0.4), likely because only specific label transitions occur.
- Normalization (NCE, NFL) offers some robustness (Fig. 6c) but is clearly outperformed by APL methods. Normalizing robust losses (MAE/RCE) shows mixed effects (Fig 6d), with NMAE underperforming MAE and NRCE offering no significant benefit over RCE at $\eta = 0.4$.



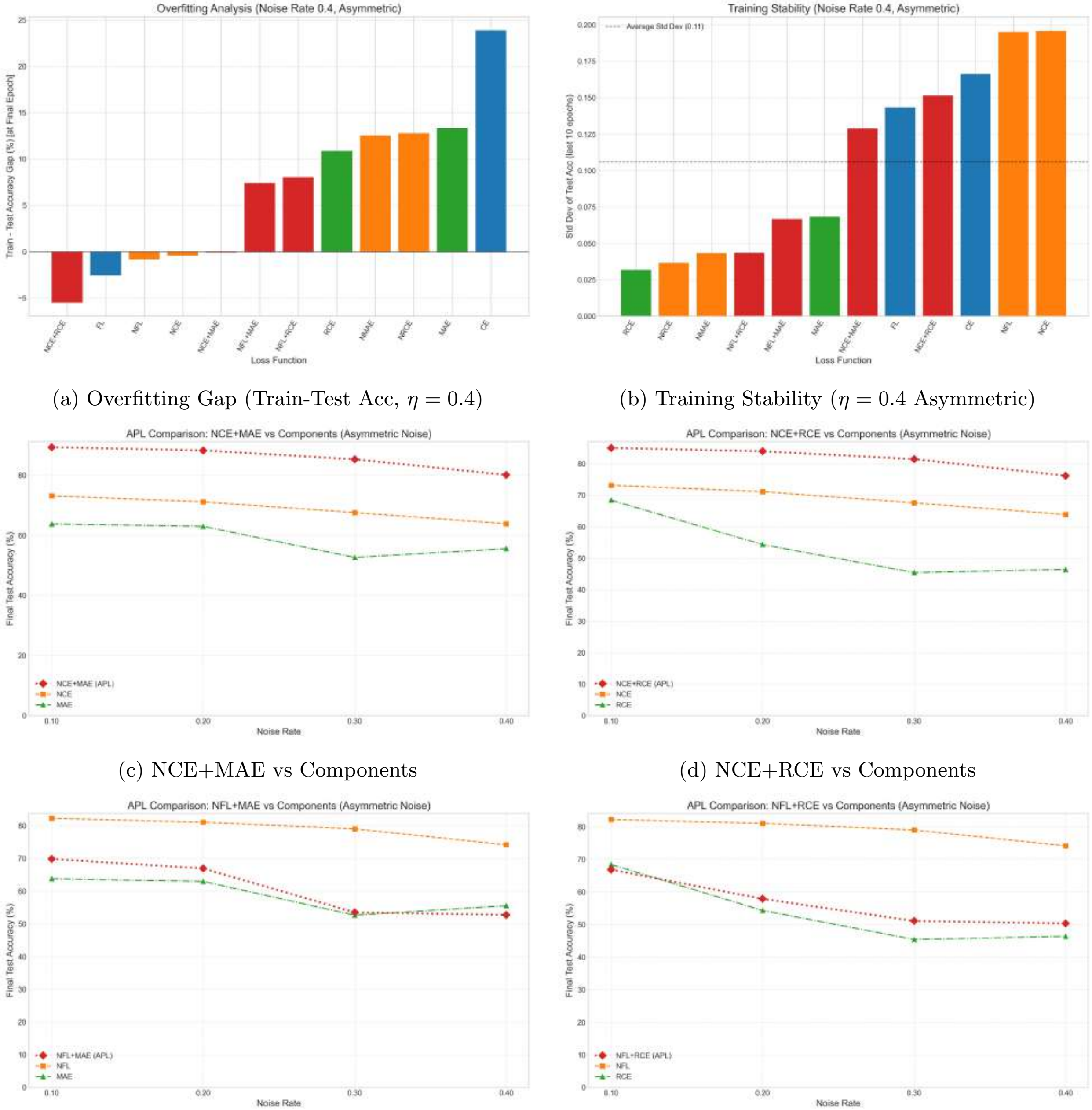
(a) Heatmap of final test accuracy (%)

(b) Final test accuracy vs. noise rate

(c) Normalization Effect (CE/FL, $\eta = 0.4$)

(d) Normalization Effect (MAE/RCE, $\eta = 0.4$)

Figure 6: Asymmetric noise performance visualizations (Part 1). (a) Heatmap. (b) Accuracy vs. noise. (c) Normalization (Std, $\eta = 0.4$). (d) Normalization (Robust, $\eta = 0.4$).



(a) Overfitting Gap (Train-Test Acc, $\eta = 0.4$)

(b) Training Stability ($\eta = 0.4$ Asymmetric)

(c) NCE+MAE vs Components

(d) NCE+RCE vs Components

(e) NFL+MAE vs Components

(f) NFL+RCE vs Components

Figure 7: Asymmetric noise performance visualizations (Part 2). (a) Overfitting gap ($\eta = 0.4$). (b) Training stability ($\eta = 0.4$). (c-f) APL synergy vs components.

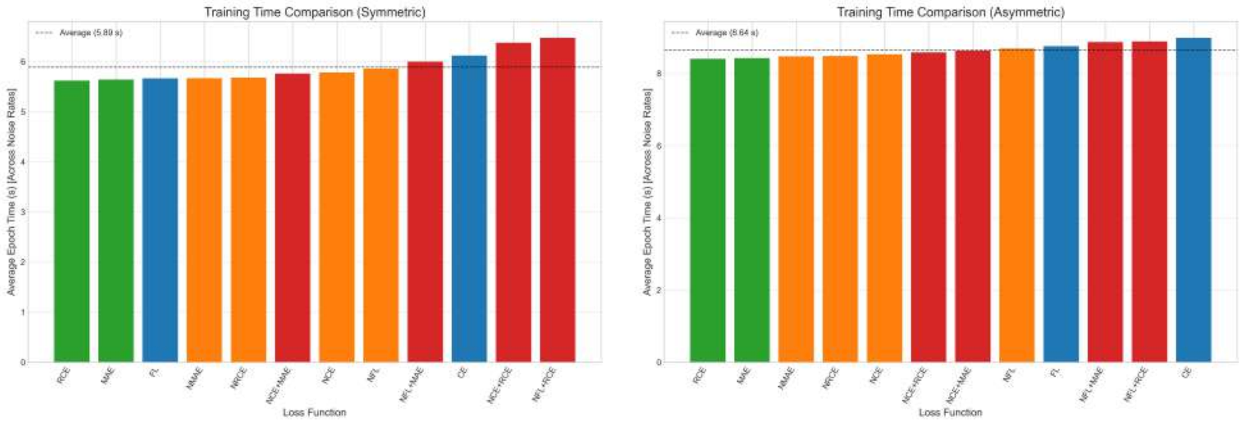
- APL methods, specifically NCE+MAE and NCE+RCE, consistently achieve top performance. NCE+MAE is top-2 across all noise rates. NCE+RCE is top-2 at $\eta = 0.3$ and $\eta = 0.4$. Standard CE is competitive at lower noise rates ($\eta = 0.1, 0.2$). The synergy plots (Fig. 7c, 7d, 7e, 7f) show APL combinations generally outperforming individual components, although the margin varies, and NCE/NFL alone are sometimes better than the passive components MAE/RCE, especially at lower noise rates.

- Similar to the symmetric case, APL methods like NCE+MAE, NCE+RCE demonstrate moderate stability during training alongside FL, while NFL+MAE, NFL+RCE demonstrate good stability alongside RCE, NRCE, and NMAE (Fig. 7b). Standard CE, NCE, and NFL exhibit higher variance in the final epochs at $\eta = 0.4$.

Detailed learning curves are in Appendix A.5.

4.4 Training Efficiency

We analyze the average training time per epoch for each loss function under both symmetric and asymmetric noise conditions, as shown in Figure 8a and 8b respectively.



(a) Symmetric Noise Training Times

(b) Asymmetric Noise Training Times

Figure 8: Comparison of average training time per epoch (seconds) across different loss functions for (a) Symmetric and (b) Asymmetric noise scenarios.

Observations (Symmetric Noise, Fig. 8a): The average epoch time across all losses is approximately 5.89 seconds.

- *Fastest:* Robust baseline losses RCE and MAE are the most efficient (≈ 5.6s/epoch).

- *Comparable to Baseline:* Focal Loss (FL) and several normalized/APL variants (NMAE, NRCE, NCE+MAE, NCE, NFL) cluster closely, taking around 5.7-5.9 seconds per epoch. Standard CE and NFL+MAE are slightly slower (≈ 5.9-6.1s).

- *Slowest:* The APL combinations involving RCE (NCE+RCE and NFL+RCE) are notably the slowest in this setup, taking around 6.3-6.4 seconds per epoch, longer than CE or standalone RCE.

Observations (Asymmetric Noise, Fig. 8b): The average epoch time is considerably higher at approximately 8.64 seconds, likely due to the absence of AMP optimization during these runs.

- *Fastest:* RCE and MAE remain the most efficient (≈ 8.4s/epoch).

- *Tightly Clustered Group:* Most other losses, including standard CE, FL, all normalized variants, and all APL combinations, fall within a relatively narrow band (approx. 8.45s to 8.8s).

- *Slowest:* Standard CE is the slowest ($\approx$ 8.8s/epoch) in this specific configuration. Unlike the symmetric case, the RCE-based APL combinations (NCE+RCE, NFL+RCE) are not the slowest; they are comparable to or slightly faster than CE.

Overall Summary: The findings show variability depending on the noise type and potentially the optimization settings (like AMP usage). However, a consistent trend is that baseline robust losses RCE and MAE are computationally efficient. Many advanced normalized and APL losses (NMAE, NRCE, NCE, NFL, NCE+MAE, NFL+MAE) offer competitive training times, often similar to or even faster than standard CE, especially under asymmetric noise conditions without AMP. The specific combinations NCE+RCE and NFL+RCE showed higher overhead in the symmetric (AMP enabled) runs but were comparable to others in the asymmetric (AMP disabled) runs, indicating their computational cost might be context-dependent but doesn't necessarily represent a prohibitive bottleneck compared to CE in all scenarios. Overall, adopting many noise-robust techniques, including several APL variants, does not inherently imply a major increase in training duration.

4.5 Analysis and Comparison with Ma et al. (2020)

Our empirical results on CIFAR-10 align well with the core findings and framework presented by Ma et al. [6]. In this context, we consider **overfitting** as the model memorizing noisy labels, leading to a large positive train-test accuracy gap, while **underfitting** refers to the failure of robust methods to achieve high absolute test accuracy, even if noise memorization is controlled.

Validation of Normalization: The experiments clearly show that applying the normalization principle (Eq. 1) transforms noise-sensitive losses (CE, FL) into more robust versions (NCE, NFL) that resist overfitting, as seen in the learning curves under high noise (e.g., Fig 3c, 6c). The overfitting analysis plots (Figs. 4a, 7a) further quantify this: NCE and NFL show significantly smaller positive overfitting gaps compared to CE and FL under high symmetric noise, and even achieve negative gaps (test accuracy > train accuracy) under moderate asymmetric noise ($\eta = 0.4$), indicating effective overfitting control. This provides practical validation for their theoretical assertion.

Robustness vs. Underfitting Trade-off: Consistent with Ma et al.'s motivation for APL, we observe that while normalized losses (NCE, NFL) and baseline robust losses (MAE, RCE) mitigate overfitting compared to CE/FL, they often achieve lower peak accuracy than the best APL methods (Tables 1, 2). This performance gap, particularly noticeable at higher noise rates, empirically demonstrates the underfitting potential of relying solely on robustness. For instance, NCE achieves reasonable robustness but lags significantly behind NCE+MAE at $\eta = 0.8$ symmetric noise and $\eta = 0.4$ asymmetric noise. The overfitting plots also suggest NMAE/NRCE might underfit more severely than MAE/RCE in some cases (e.g., larger positive gap for NMAE vs MAE at $\eta = 0.4$ asymmetric).

APL Effectiveness and Synergy: The APL framework demonstrates remarkable effectiveness. The combinations of a robust active loss (NCE or NFL) and a robust passive loss (MAE or RCE) consistently rank among the top performers across nearly all noise conditions, both symmetric and asymmetric. Crucially, the APL combinations frequently outperform both of their constituent components when used alone (e.g., Fig 5a, 7d, 7e, 7f). Furthermore, the overfitting analysis (Figs. 4a, 7a) reveals interesting dynamics. Under high symmetric noise ($\eta = 0.8$), the APL methods exhibit substantial *negative* overfitting gaps, suggesting the test accuracy surpasses the final training accuracy, possibly indicating a strong regularization effect or difficulty in fitting the extremely noisy training data perfectly. In contrast, under moderate asymmetric noise ($\eta = 0.4$), NCE+RCE shows a negative gap, while NCE+MAE and NFL+MAE achieve small positive gaps, indicating excellent overfitting control compared to the large positive gap of CE. This clearly illustrates the synergy between active and passive components in the APL, where the active component provides a strong learning signal, the passive component enhances robustness and discrimination, and the combination leads to a better overall balance of accuracy and overfitting control. Our results strongly support APL as a superior strategy.

Specific Loss Performance:

- *NCE+MAE:* Consistently strong accuracy (top-2 symmetric and asymmetric). Exhibits excellent overfitting control: large negative gap at $\eta = 0.8$ symmetric, near-zero gap at $\eta = 0.4$ asymmetric.
- *NCE+RCE:* Excellent accuracy, especially under asymmetric noise (top-2 at $\eta = 0.3, 0.4$). Shows strong overfitting control with negative gaps in both high-noise scenarios plotted.
- *NFL+RCE:* Very strong accuracy under symmetric noise (top-2 at $\eta = 0.6, 0.8$). Shows large negative overfitting gap at $\eta = 0.8$ symmetric, but a moderate positive gap at $\eta = 0.4$ asymmetric. Less effective accuracy-wise under asymmetric noise compared to NCE-based APL.
- *NFL+MAE:* Mixed accuracy results. Shows large negative overfitting gap at $\eta = 0.8$ symmetric, small positive gap at $\eta = 0.4$ asymmetric.
- *Baselines:* CE is optimal on clean data, competitive at low asymmetric noise, but shows the largest positive overfitting gap under high noise. FL also shows significant positive overfitting under symmetric noise, but surprisingly controls it well (negative gap) under $\eta = 0.4$ asymmetric noise despite lower accuracy than APL. MAE/RCE show intermediate positive overfitting gaps.
- *Normalized Baselines:* NCE/NFL control overfitting much better than CE/FL (moderate positive gap symmetric, negative gap asymmetric) but underperform APL combinations significantly in accuracy. NMAE/NRCE generally show moderate positive overfitting gaps, not consistently better than MAE/RCE in this regard, and underperform significantly in accuracy.

These specific performance characteristics align with the general trends reported in [6], although relative rankings can vary slightly based on architecture and use of performance enhancing techniques such as AMP. Our use of fixed $\alpha = \beta = 1$ provides a strong baseline validation; further tuning could potentially optimize specific APL pairs for particular noise settings.

5 Limitations and Future Directions

While this study provides valuable empirical insights into loss function normalization and the APL framework on CIFAR-10, several limitations suggest avenues for future research:

- **Dataset and Architecture Scope:** The primary evaluation was conducted on CIFAR-10 using a specific 8-layer CNN. Validating these findings on larger-scale datasets (e.g., ImageNet, WebVision full set) and with diverse, deeper architectures (like wider ResNets or Vision Transformers) is crucial to confirm the generalizability of APL's effectiveness.
- **APL Parameter Sensitivity:** We used fixed, equal weights ($\alpha = \beta = 1$) for the APL combinations for simplicity and baseline validation. As hinted by the original APL paper [6] and suggested by the differing optimal parameters found for CIFAR-10 vs. CIFAR-100 in that work, performance could potentially be further optimized by tuning α and β. Future work could investigate methods for adaptive or dataset/noise-specific parameter selection, perhaps even dynamically during training.
- **Complexity of Real-World Noise:** The symmetric and asymmetric noise models, while standard benchmarks, may not fully capture the intricacies of noise in real datasets, which can be instance-dependent or feature-dependent. Evaluating APL under these more complex noise scenarios would be valuable.
- **Optimization Dynamics:** The observed underperformance of some theoretically robust normalized losses (NMAE, NRCE) suggests that factors beyond theoretical robustness, such as optimization landscape or gradient properties, play a significant role. Further investigation into why certain combinations excel while others falter during optimization is warranted.
- **Comparison with Other Paradigms:** This study focused on comparing different loss functions. A broader comparison including state-of-the-art sample selection methods or label correction techniques could provide further context on the relative strengths of the APL approach within the wider field of learning with noisy labels.

- **Hybrid Approaches:** Exploring combinations of the APL framework with other techniques, such as integrating sample selection heuristics or semi-supervised learning elements, could potentially lead to even more robust and accurate models.

Addressing these points will help to further solidify the understanding and practical application of normalized losses and the APL framework in diverse deep learning scenarios affected by label noise.

6 Conclusion

This study empirically evaluated the effectiveness of loss function normalization and the Active Passive Loss (APL) framework, inspired by Ma et al. [6], for training CNNs on CIFAR-10 in the presence of both symmetric and asymmetric label noise. Our findings strongly corroborate the key theoretical insights: (1) Normalization effectively induces robustness in standard loss functions like CE and FL, significantly reducing overfitting compared to their non-normalized counterparts. (2) Relying solely on robustness (e.g., using only NCE, NFL, MAE, or RCE) can lead to underfitting, limiting the model's ability to achieve optimal accuracy, particularly as noise increases. (3) The APL framework, by combining complementary robust active and passive loss components, successfully addresses this trade-off, consistently achieving superior performance and exhibiting excellent overfitting control (often resulting in small positive or negative train-test accuracy gaps under high noise), especially under challenging high-noise conditions.

Specifically, APL methods like NCE+MAE and NCE+RCE demonstrated state-of-the-art accuracy results within our experimental setup across various noise types and rates, significantly outperforming standard CE/FL, baseline robust losses MAE/RCE, and their normalized counterparts NCE/NFL/NMAE/NRCE. The often competitive computational overhead associated with many of these methods further enhances their practical appeal. This work provides robust empirical evidence supporting the APL framework as a principled, effective, and generally efficient design strategy for mitigating the detrimental effects of noisy labels, balancing noise tolerance, learning capacity, and overfitting control in deep learning classification tasks.

References

[1] Aritra Ghosh, Himanshu Kumar, and PS Sastry. Robust loss functions under label noise for deep neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017. doi: 10.1609/aaai.v31i1.10894. URL https://doi.org/10.1609/aaai.v31i1.10894.

[2] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in Neural Information Processing Systems*, 2018. URL https://arxiv.org/abs/1804.06872.

[3] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*. PMLR, 2018. URL https://proceedings.mlr.press/v80/jiang18c.html.

[4] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

[5] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017. doi: 10.1109/ICCV.2017.324. URL https://doi.org/10.1109/ICCV.2017.324.

[6] Xingjun Ma, Hanxun Huang, Yisen Wang, Simone Romano, Sarah Erfani, and James Bailey. Normalized loss functions for deep learning with noisy labels. In *International Conference on Machine Learning*. PMLR, 2020. URL https://proceedings.mlr.press/v119/ma20c.html.

[7] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach: An oral paper at: Cvpr. 2017. URL https://arxiv.org/abs/1609.03683.

[8] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014. URL https://arxiv.org/abs/1409.1556. arXiv preprint.

[9] Arash Vahdat. Toward robustness against label noise in training deep discriminative neural networks. In *Advances in Neural Information Processing Systems*, 2017. URL https://arxiv.org/abs/1706.00038.

[10] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *ICCV*, 2019. URL https://arxiv.org/abs/1908.06112.

[11] Zhilu Zhang and Mert R Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in Neural Information Processing Systems*, 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/f2925f97bc13ad2852a7a551802feea0-Paper.pdf.

A Appendix

A.1 CNN Model Architecture Details

The CNN architecture (`CNN_Model`) used is a sequential model designed for 32x32 input images, visualized in Figure 9. It follows a standard VGG-like pattern with Batch Normalization (BN) and ReLU activations.

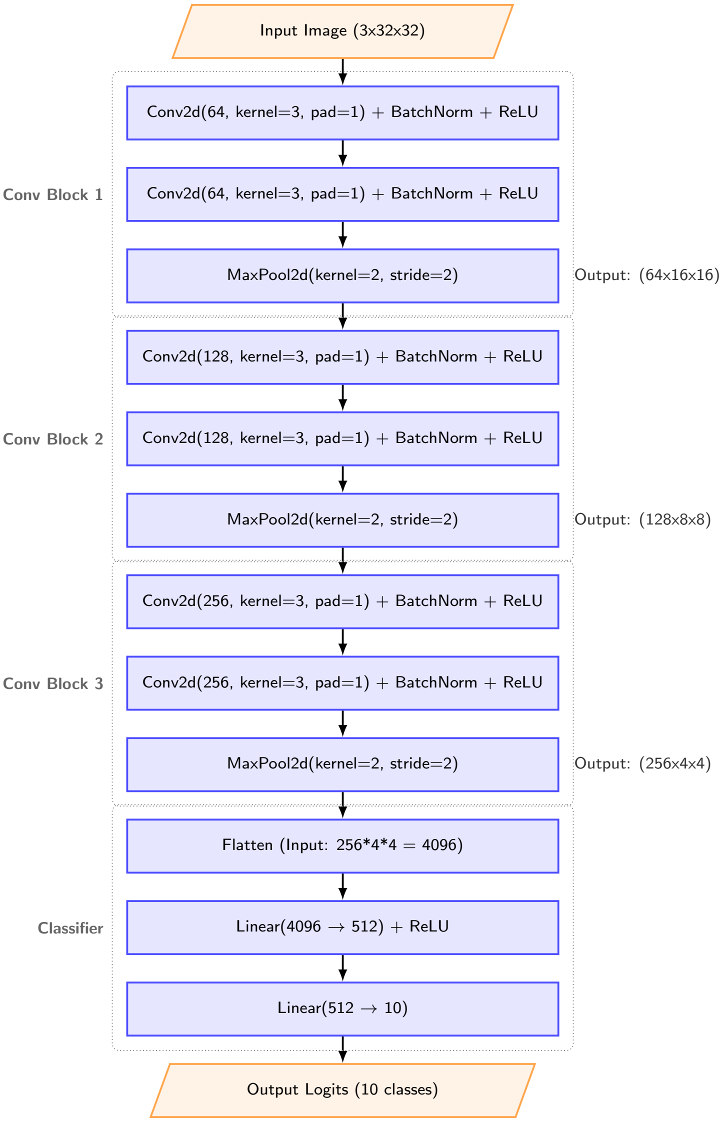

Input Image (3x32x32)
Conv Block 1
Conv2d(64, kernel=3, pad=1) + BatchNorm + ReLU
Conv2d(64, kernel=3, pad=1) + BatchNorm + ReLU
MaxPool2d(kernel=2, stride=2)
Output: (64x16x16)
Conv Block 2
Conv2d(128, kernel=3, pad=1) + BatchNorm + ReLU
Conv2d(128, kernel=3, pad=1) + BatchNorm + ReLU
MaxPool2d(kernel=2, stride=2)
Output: (128x8x8)
Conv Block 3
Conv2d(256, kernel=3, pad=1) + BatchNorm + ReLU
Conv2d(256, kernel=3, pad=1) + BatchNorm + ReLU
MaxPool2d(kernel=2, stride=2)
Output: (256x4x4)
Classifier
Flatten (Input: 256*4*4 = 4096)
Linear(4096 → 512) + ReLU
Linear(512 → 10)
Output Logits (10 classes)

Figure 9: CNN architecture diagram illustrating convolutional blocks and the final classifier. Each Conv Block consists of two Conv2d layers (with BatchNorm and ReLU) followed by MaxPool2d.

A.2 Asymmetric Noise Mapping Details

The specific class transitions for asymmetric noise are:

- TRUCK → AUTOMOBILE
- BIRD → AIRPLANE
- DEER → HORSE
- CAT → DOG
- DOG → CAT

Labels for Frog and Ship classes remain unchanged (clean), and only CAT ↔ DOG is bidirectional.

A.3 Normalized Loss Simplifications

The simplified forms for NMAE and NRCE used in Section 3.4 and implemented in the code arise from applying the normalization definition (Eq. 1). For NMAE, with $l_{MAE}(j) = 2(1 - p_j)$, the normalization is $\frac{2(1-p_y)}{\sum_j 2(1-p_j)} = \frac{1-p_y}{K-1}$. For NRCE, with $l_{RCE}(j) = -A(1 - p_j)$, the normalization yields $\frac{-A(1-p_y)}{-A(K-1)} = \frac{1-p_y}{K-1}$. Note that while mathematically equivalent after normalization in this specific case (using the simplified RCE term), their gradients and numerical behavior might differ slightly in practice due to the underlying base loss definitions. The code implementations `NMAELoss` and `NRCELoss` use these simplified forms (with potential scaling factors as noted before).

A.4 Symmetric Noise Learning Curves

Figures 10-14 show Train/Test Accuracy vs. Epoch for different loss categories under symmetric noise.

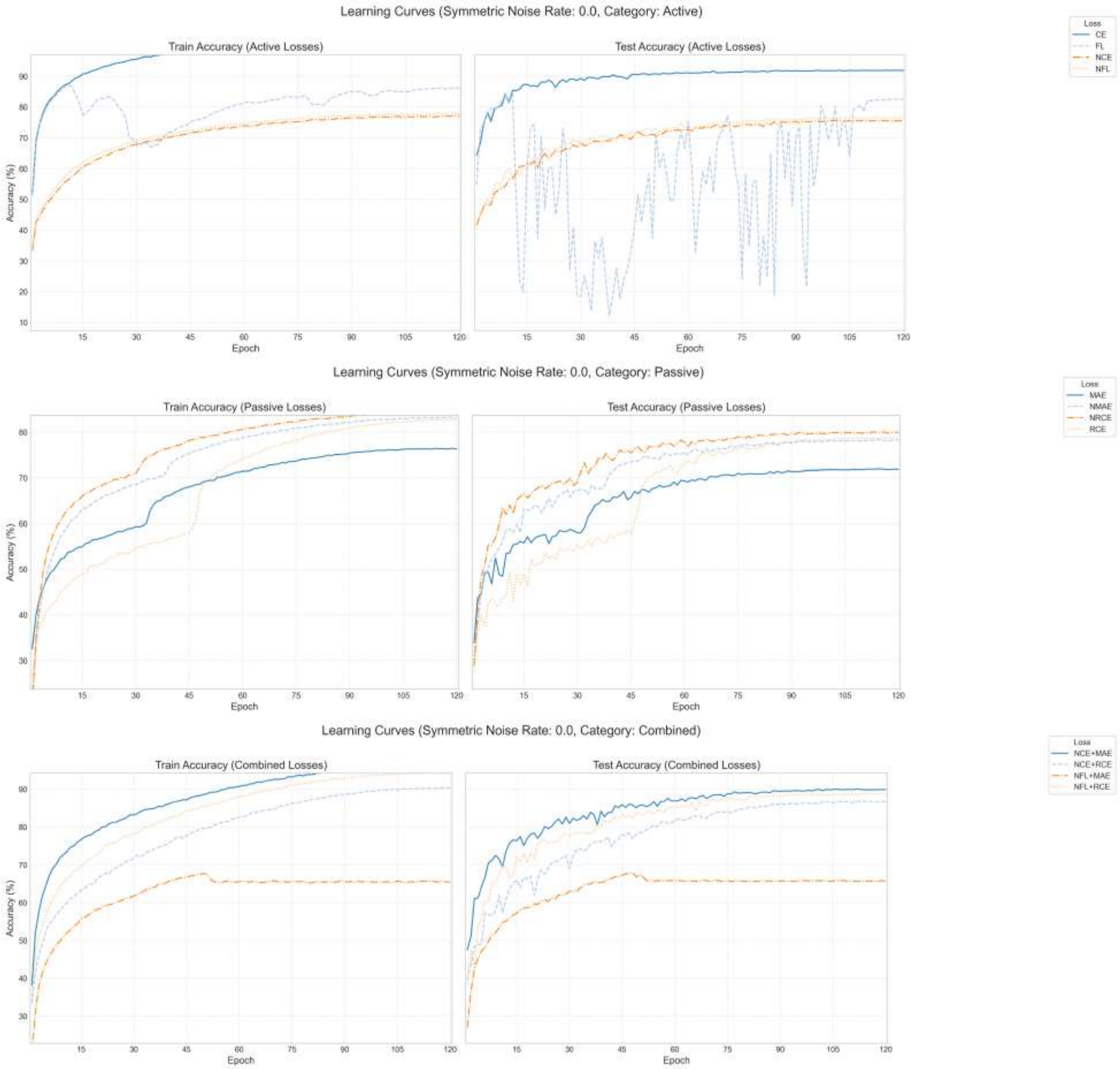


Figure 10: Symmetric Noise: Learning Curves ($\eta = 0.0$, Clean)

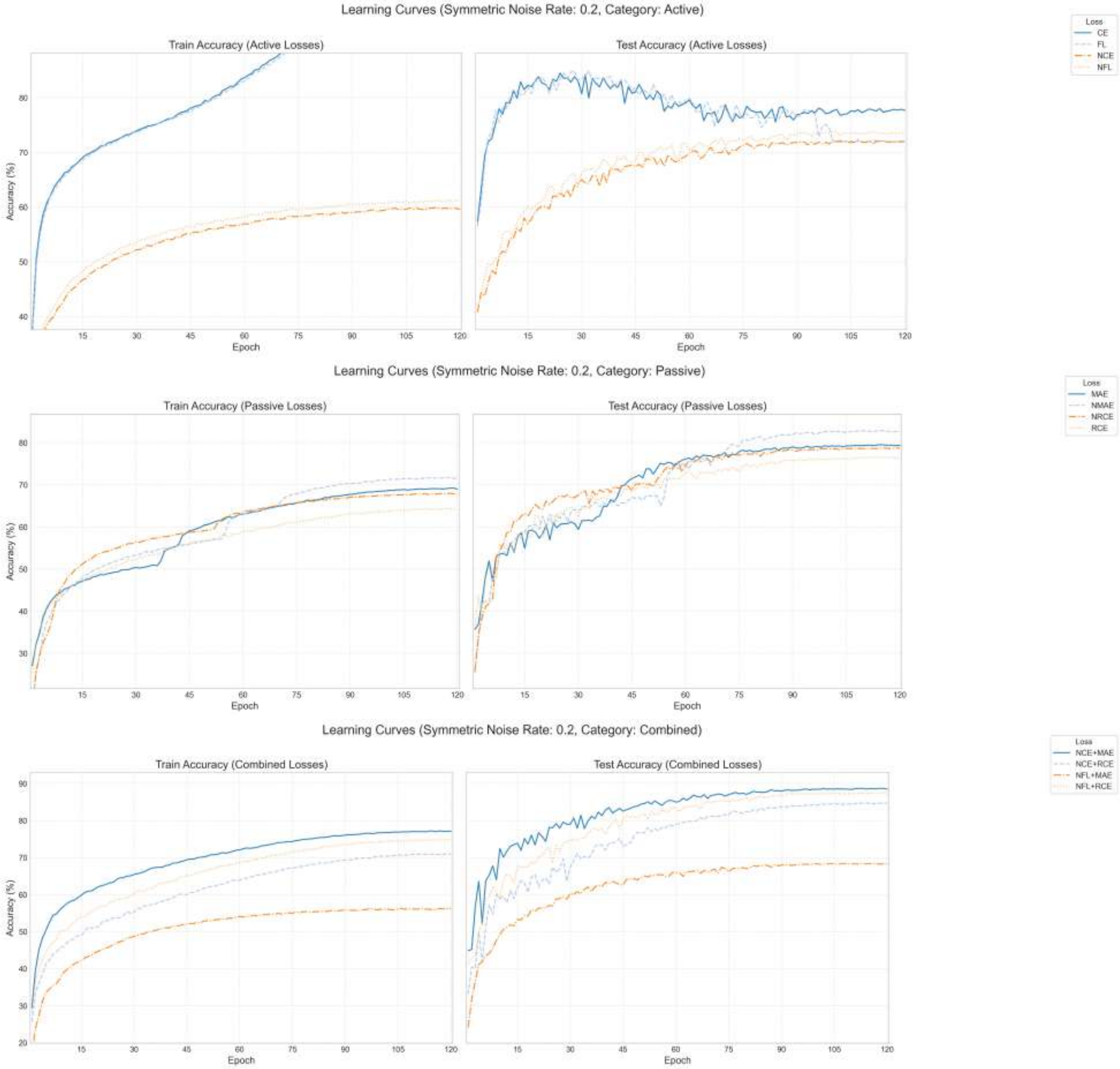


Figure 11: Symmetric Noise: Learning Curves ($\eta = 0.2$)

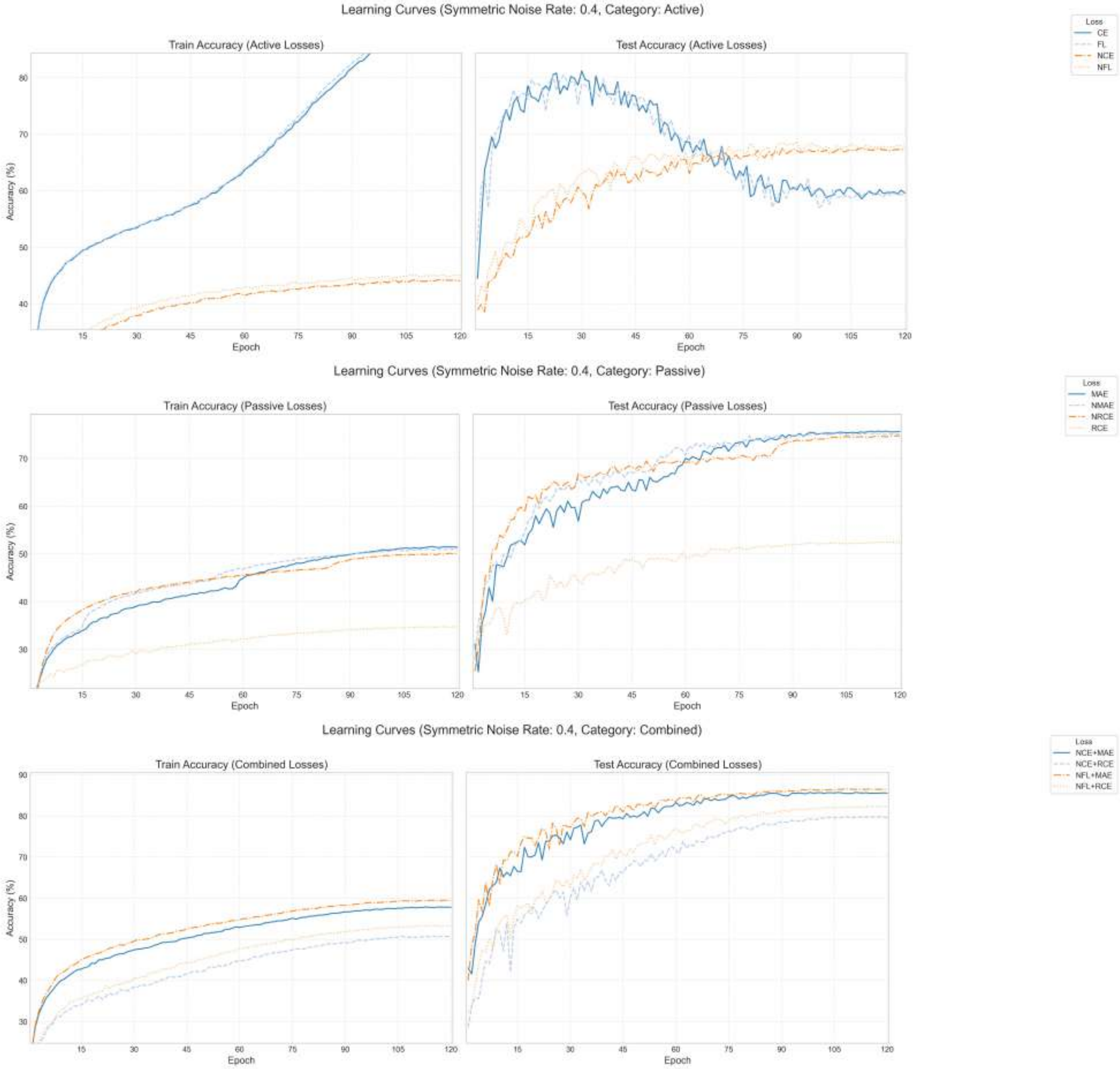


Figure 12: Symmetric Noise: Learning Curves ($\eta = 0.4$)

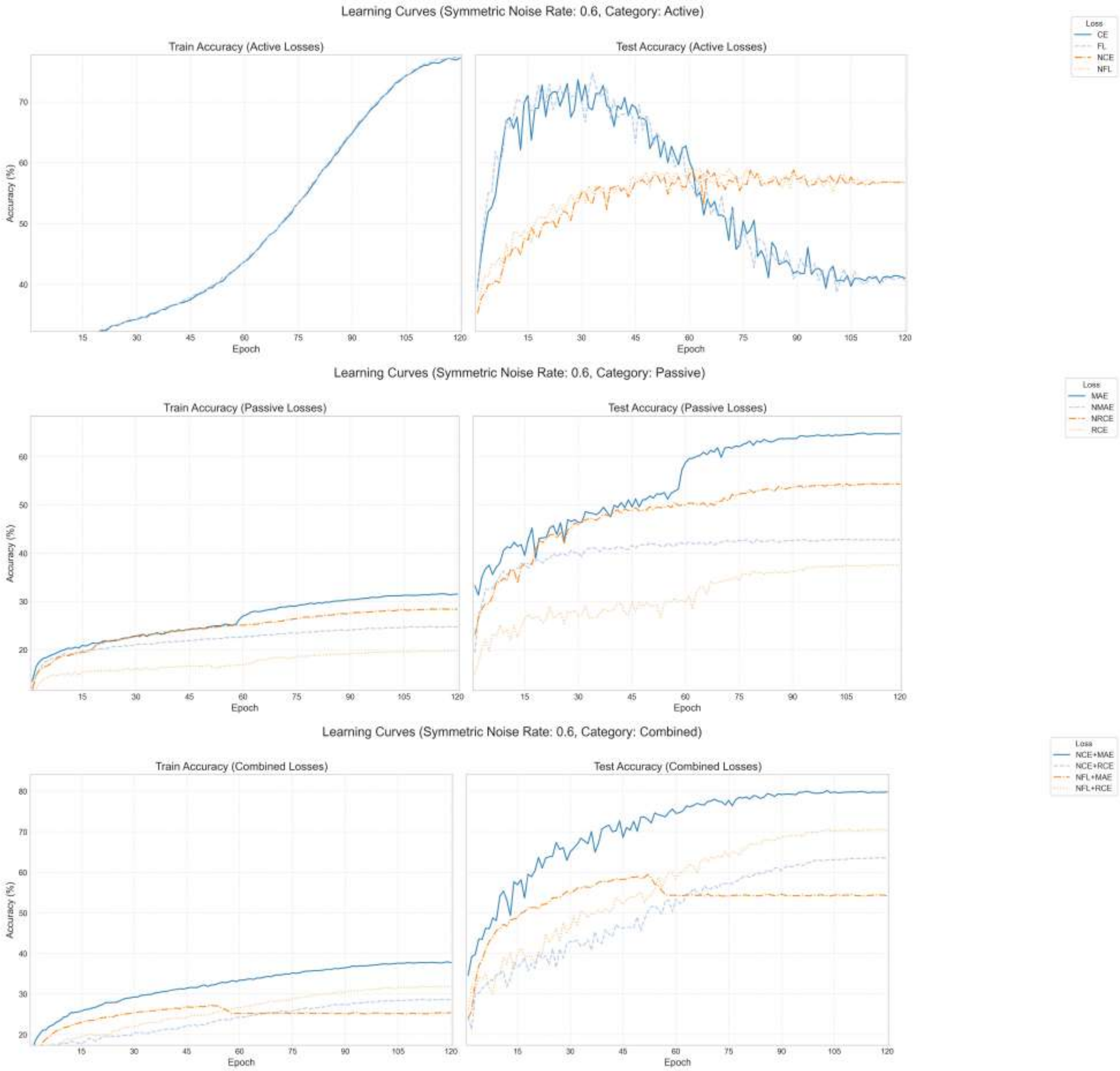


Figure 13: Symmetric Noise: Learning Curves ($\eta = 0.6$)

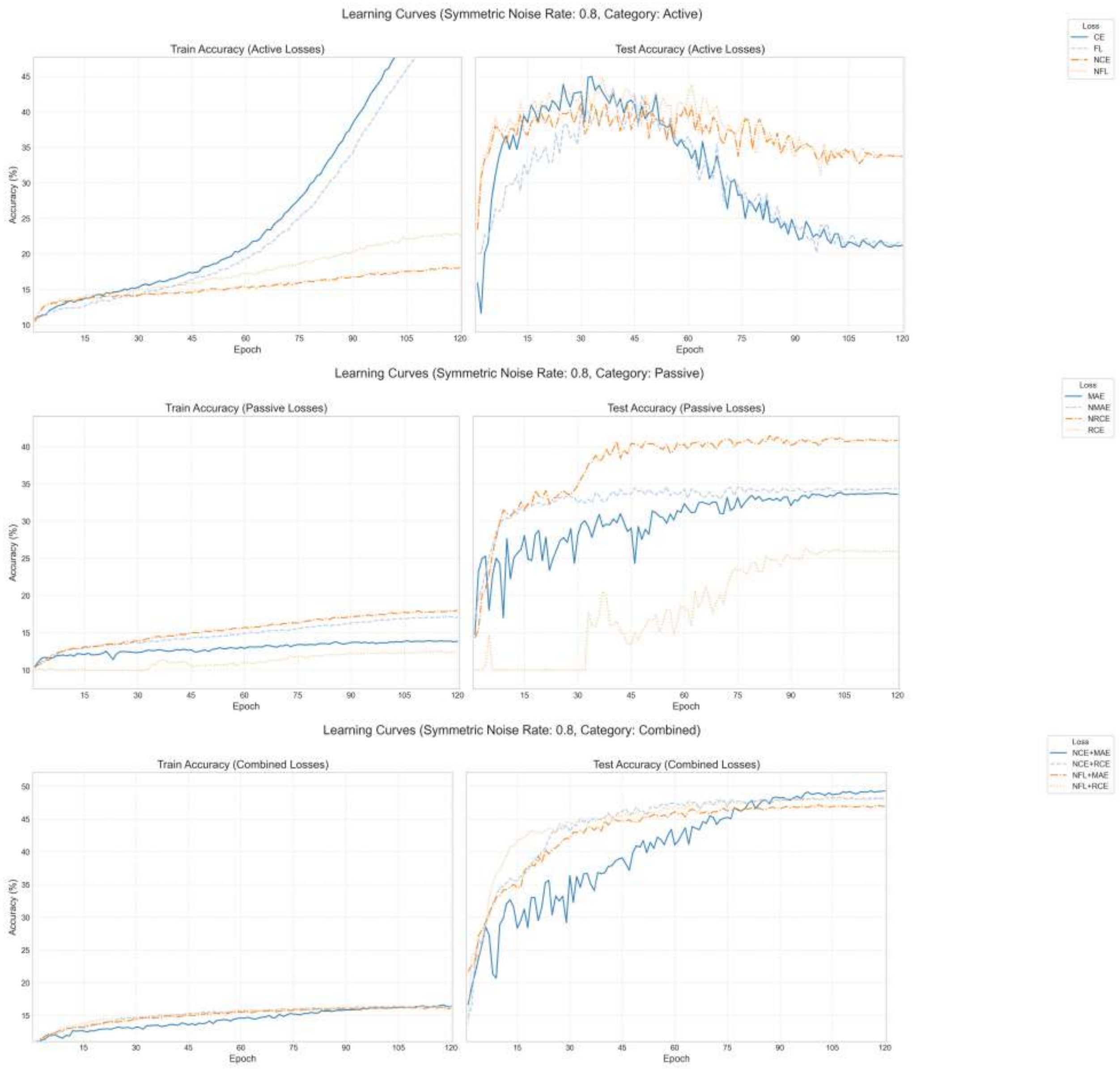


Figure 14: Symmetric Noise: Learning Curves ($\eta = 0.8$)

A.5 Asymmetric Noise Learning Curves

Figures 15-18 show Train/Test Accuracy vs. Epoch for different loss categories under asymmetric noise.

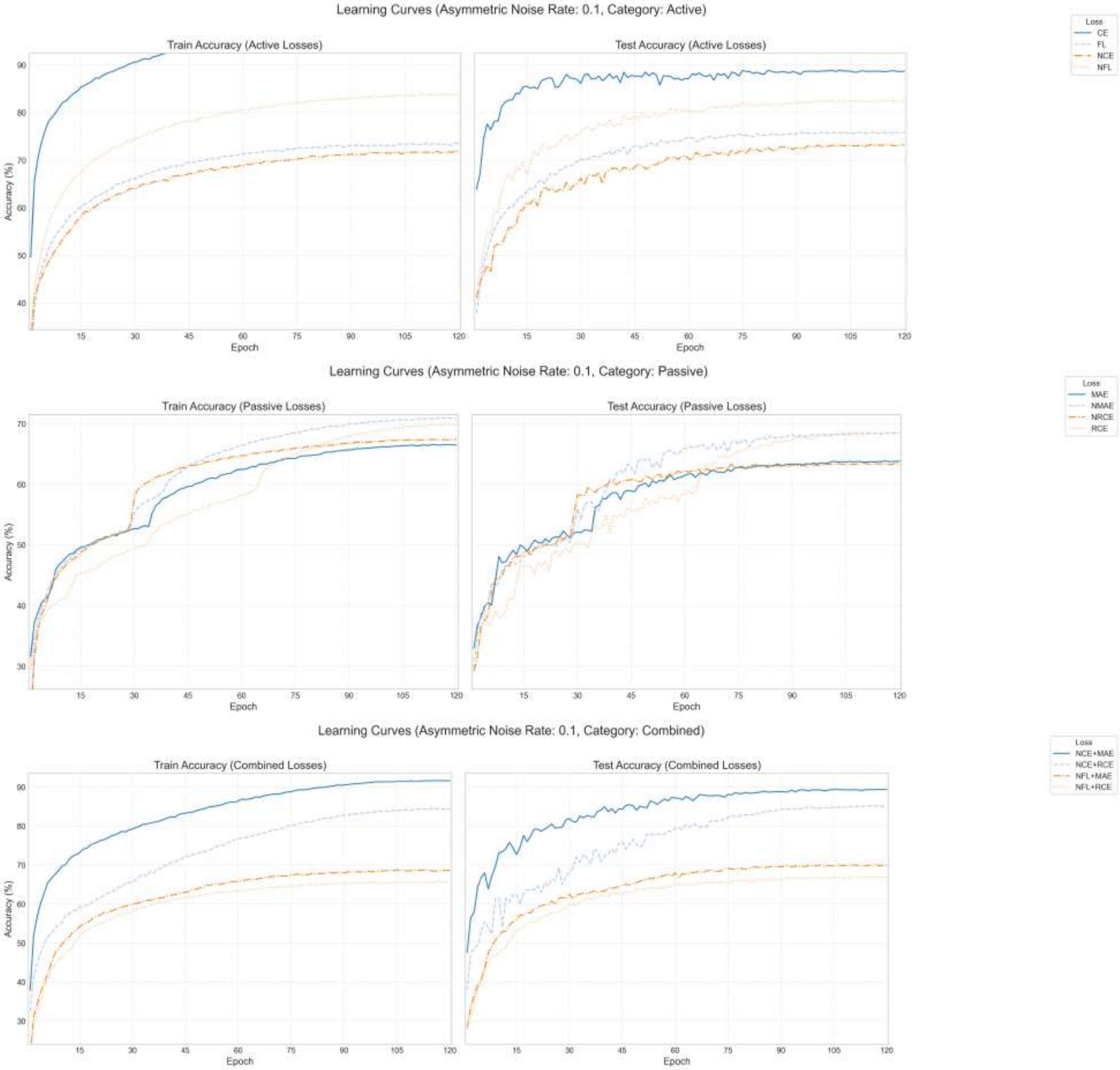


Figure 15: Asymmetric Noise: Learning Curves ($\eta = 0.1$)

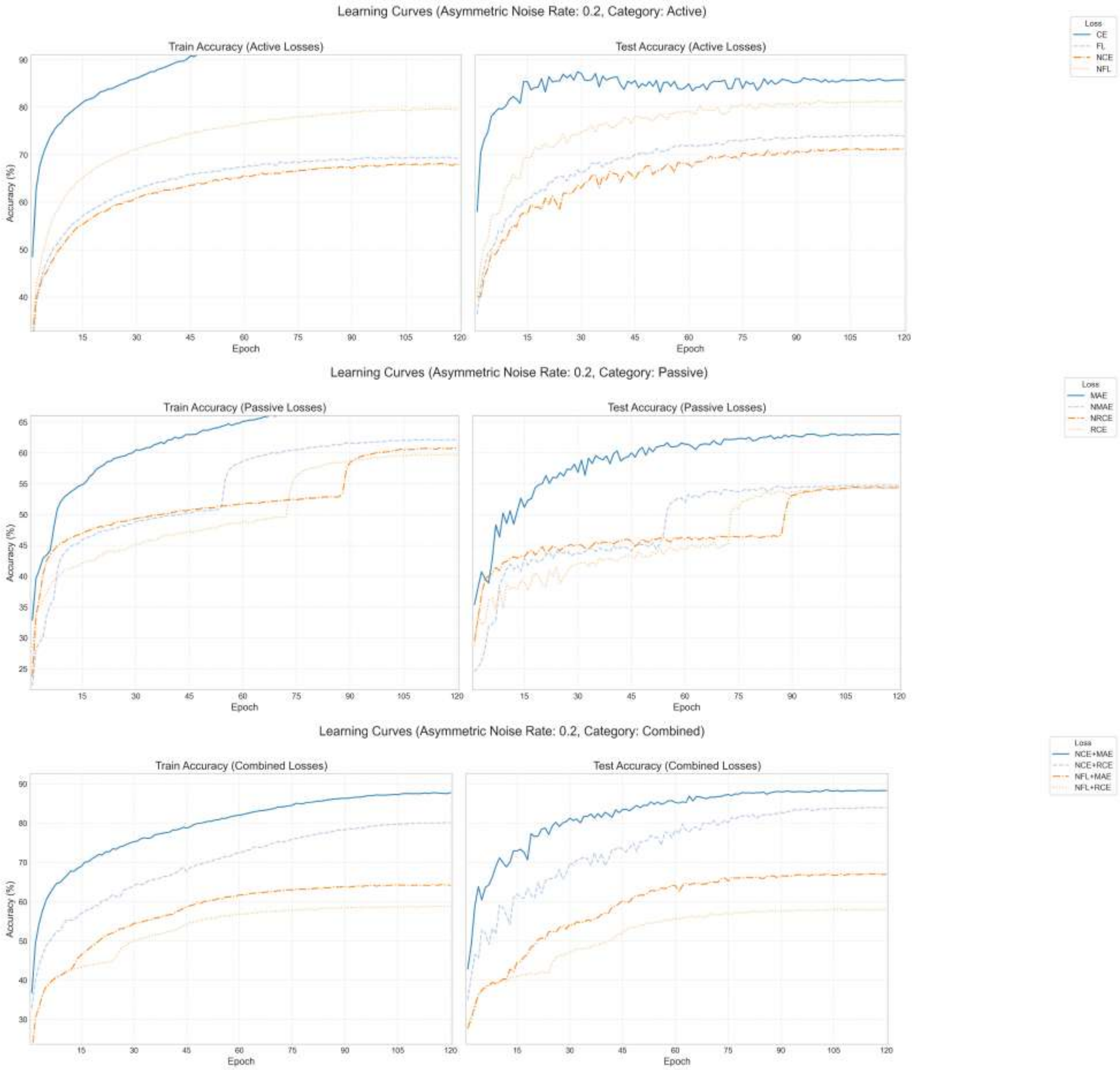


Figure 16: Asymmetric Noise: Learning Curves ($\eta = 0.2$)

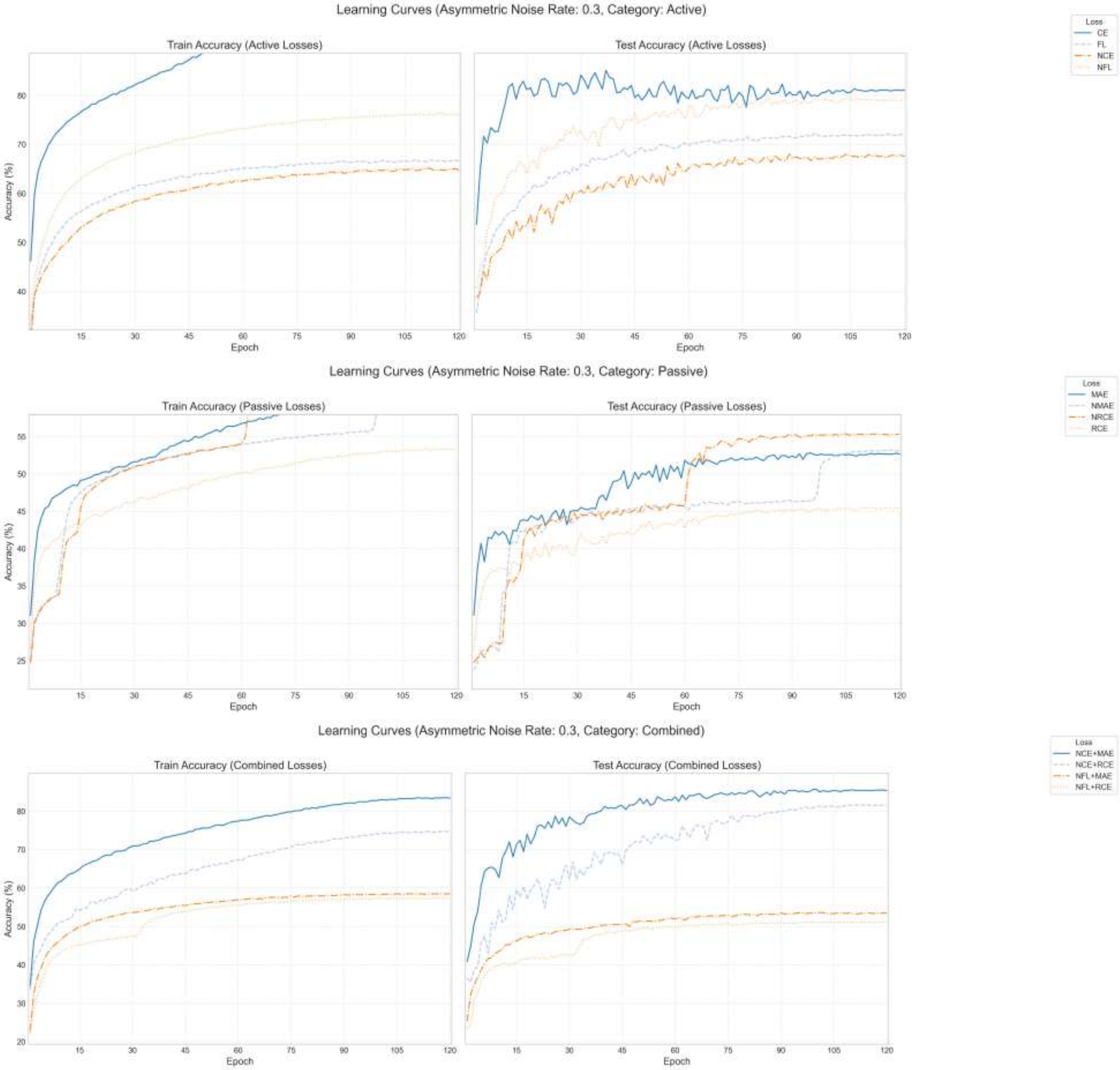


Figure 17: Asymmetric Noise: Learning Curves ($\eta = 0.3$)

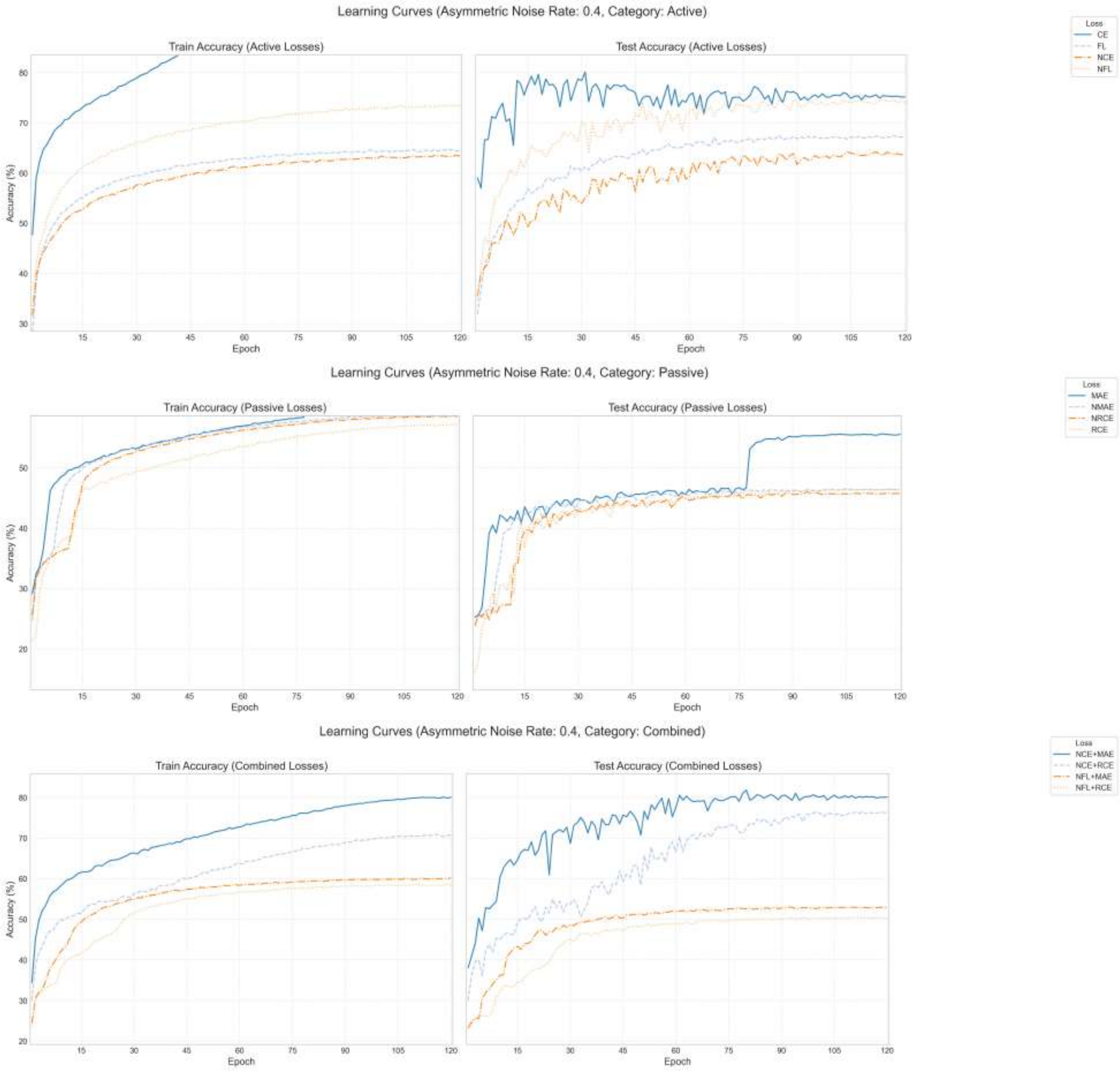


Figure 18: Asymmetric Noise: Learning Curves ($\eta = 0.4$)