DRAMoE: Boosting Adversarial Robustness with Adversarial Training and Adaptive Mixture of Experts

Yu Fu$^{1[0009-0003-2234-7312]}$, Hengzhi Xie$^{1[0009-0009-8871-940X]}$, Ming Yang^{2}, and Denghui Zhang$^{1,2(✉)}$

[1] Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou, China
denghui.zhang@gzhu.edu.cn

[2] Qilu University of Technology, Shandong, China
yangm@sdas.org

Abstract. Deep learning (DL) has been poised to become an integral part of many critical systems. Despite its robustness to natural variations, it is highly susceptible to adversarial examples (AEs) generated by subtle, imperceptible perturbations. As a standard defense form, Adversarial Training (AT) enhances robustness by incorporating AEs during training, yet the distributional gap between clean examples and AEs often compromises standard accuracy, posing a trade-off challenge. By routing examples to matching experts, the Mixture of Experts (MoE) offers a solution. However, current MoE-based AT methods are limited by global pooling-based routers and fail to exploit routing capabilities fully during the AT. This work proposes DRAMoE (Dynamic Router-guided Adversarial Mixture of Experts), a novel MoE architecture integrating AT with a divide-and-conquer strategy. DRAMoE comprises two core components: a Dynamic Channel Attention Router (DCAR) and Router-Guided Adversarial Training (RGAT). DCAR enhances feature representation, while RGAT optimizes AEs generation and designs diverse losses. Extensive experiments on CIFAR-10, CIFAR-100, and Tiny ImageNet demonstrate that DRAMoE improves robust accuracy over the baseline while maintaining competitive standard accuracy.

Keywords: Adversarial Defense · Mixture of Experts · Adversarial Training · AI Security · Robustness

1 Introduction

DL has achieved remarkable achievements in fields such as autonomous driving, smart healthcare, and Internet finance [16]. Its ability to automatically learn and generalize features has significantly promoted the rapid development of artificial intelligence. However, their black-box nature and inherent vulnerabilities make them susceptible to AEs [22, 6]. The imperceptible AEs can mislead models into erroneous decisions with high confidence, threatening model security [9]. Many studies have focused on defending neural networks against such adversarial attacks. Current defense strategies primarily fall into two categories: input transformation and AT. Input transformation including feature compression [31], frequency-domain transformation [34], and data dropping [7], preprocess inputs to mitigate adversarial perturbations through dimensionality reduction, frequency filtering, or selective removal. However, these methods are limited by gradient obfuscation [2] and adaptive attacks like AutoAttack [5], and by high computational costs unsuitable for real-time applications, constraining their comprehensive defense capability. In contrast, another class of defense methods, AT, optimizes model parameters by incorporating AEs during training [18], which has become a standard defense form for its effectiveness.

AT often struggles with distributional mismatches between clean examples and AEs, causing decision boundary shifts and reduced generalization on clean examples [25]. Subtle adversarial perturbations disrupt classification stability [30], making it challenging to balance robustness and standard accuracy [9]. Methods like TRADES [35], MART [28], and FAT [36] introduce balancing mechanisms, but distributional differences still constrain their performance.

To address this trade-off, MoE models dynamically select each expert via a router based on input distributions, effectively handling heterogeneous data [12, 21]. Leveraging sparsity and dynamic routing, MoE demonstrates strong generalization in DL models [4] and has gained attention in adversarial defense. However, existing MoE-based AT methods [38] rely on a global average pooling mechanism of routers, which leads to a focus on global features, limiting their ability to capture fine-grained input differences between AEs and clean examples, and the role of the router has not been rigorously examined within the broader context of AT.

Thus, this paper proposes DRAMoE, a Dynamic Router-guided Adversarial MoE architecture that adopts a divide-and-conquer strategy. By enhancing the role of the router in AT, DRAMoE mitigates the trade-off between standard accuracy and robustness. The contributions of this paper are as follows:

- We develop a novel AT strategy integrating MoE with a divide-and-conquer approach to balance standard accuracy and robustness.
- We first propose SRAMoE, a static MoE architecture combining pre-trained clean and adversarial experts via routing. Its limited adaptability, shown through experiments and theory, motivates a dynamic MoE design.
- We further propose DRAMoE, leveraging a dynamic MoE architecture, introducing DCAR to enhance feature representation, and adopting RGAT to optimize AE generation and design diverse losses.
- Extensive experiments on datasets validate the effectiveness and strong defense of our framework.

The paper is structured as follows: Section 2 reviews the literature on adversarial attacks, defenses, and MoE. Section 3 describes DRAMoE in detail; the architecture of DRAMoE is shown in Fig. 1. Section 4 presents the performance of DRAMoE on different datasets and attacks. Section 5 summarizes the findings and future work.

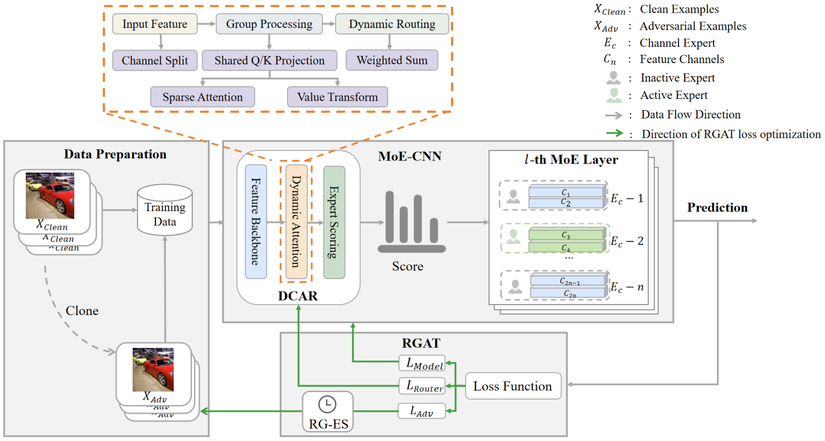


Fig. 1. Overview of DRAMoE Architecture. We propose a Dynamic Router-guided Adversarial MoE framework based on MoE-CNN [38], where the DCAR computes feature scores via grouped self-attention to activate experts. RGAT takges an early-stopping mechanism (RG-ES) to optimize AEs generation, with losses L_{Model}, L_{Router}, and L_{Adv} optimizing the model, router, and AEs, respectively, enhancing robustness and standard accuracy during training.

2 Related Work

2.1 Adversarial Attacks

The generation of AEs is often modeled as an optimization problem to find perturbations that cause the model to mispredict the inputs under specific paradigm constraints. To exploit the vulnerabilities, researchers have proposed a variety of attack strategies. Goodfellow et al. [9] introduce the Fast Gradient Sign Method (FGSM), using loss function gradients to quickly generate AEs. Kurakin et al. [14] develop the Basic Iterative Method (BIM), enhancing attack efficacy through multiple small-step iterations. Madry et al. [18] propose PGD, generating stronger AEs via multi-step optimization, serving as a white-box attack benchmark. Additionally, Croce and Hein [5] introduce AutoAttack, an ensemble framework integrating multiple attack strategies, adaptively selecting optimal methods to significantly improve attack reliability and effectiveness, widely used for evaluating model robustness.

2.2 Adversarial Training

To defend against adversarial attacks, various strategies have been developed, with AT as a primary method. AT bolsters model robustness by integrating AEs into training, using a min-max optimization to minimize loss under perturbations. Among notable AT methods, the PGD-based approach by Madry et al. [18] employs multi-step projected gradient descent to craft AEs, thereby significantly improving model robustness. MART by Wong et al. [28] optimize efficiency via kernel techniques, reducing computational costs. TRADES by Zhang et al. [35] balances robustness and standard accuracy through a dual-objective loss. Friendly Adversarial Training (FAT) by Zhang et al. [36] improves the adaptability of AEs by preserving generalization through early-stopping PGD to select low-loss AEs.

Although AT enhances model robustness, it often leads to a drop in standard accuracy. Tsipras et al. [23] theoretically demonstrate this trade-off, showing that AT encourages smoother decision boundaries that differ significantly from those learned by standard models. This observation has been further validated by Wei et al. [26], underscoring the challenge of balancing robustness and standard accuracy. To address this, we explore the use of a dynamic network to integrate a MoE architecture with AT, enabling adaptive expert selection and improving the trade-off between robustness and performance.

2.3 Mixture of Experts

MoE can enhance global performance by combining multiple experts. Unlike traditional ensemble methods [21, 3, 8, 27], MoE employs a routing mechanism that dynamically selects relevant experts and adjusts their contribution weights based on input features, improving adaptability to diverse data. The MoE framework often comprises multiple expert networks and a routing network [12], where each expert specializes in specific input features, and the router determines their involvement based on input characteristics.

MoE models can be categorized into static and dynamic architectures. Static MoE relies on pre-trained experts and fixed routing mechanisms for rapid deployment, which limits its adaptability. Dynamic MoE enhances its generalization capability through online optimization of experts and dynamic routing. For example, MoE-CNN [32, 24, 38] uses convolutional experts and sparse routing to improve efficiency and robustness but remains limited to standard training paradigms.

To optimize MoE-CNN performance, researchers have explored improvements in standard accuracy and robustness. For standard accuracy, attention mechanisms improve router feature modeling for precise expert matching [37]. For robustness, ADVMoE introduces a novel dual-layer optimization framework [38], jointly optimizing AT of the main network and the routing strategy, improving MoE-CNN robustness under adversarial attacks.

3 Methodology

In this section, we first explore a static MoE architecture, SRAMoE, and reveal the adaptability limitations of its coarse-grained experts and static fusion mechanism; then, we introduce a dynamic MoE architecture, DRAMoE, to address these limitations with fine-grained experts and dynamic routing.

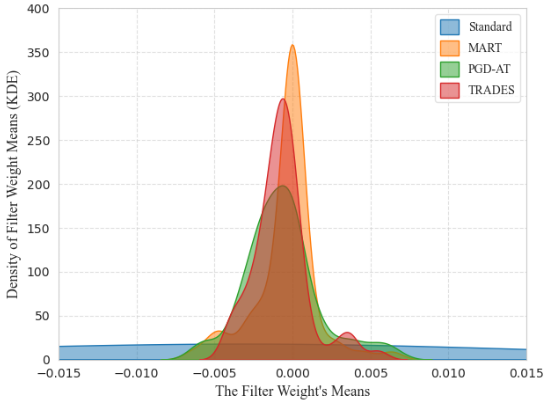


Fig. 2. Weight distribution differences in the second conv layer of ResNet-18 between a standard model and AT models (MART, PGD-AT, TRADES).

3.1 The Static Design of SRAMoE

Static MoE architectures, leveraging pre-trained experts, are a common starting point in MoE research due to their simplicity and efficiency. To explore MoE in AT, we first propose SRAMoE, which integrates a standard pre-trained expert model E_{clean}, optimized for clean examples, and an adversarial pre-trained expert model E_{adv}, optimized for AEs, to assign different data types to corresponding experts. During inference, the router G computes weights W_{clean} and W_{adv} based on input example features, producing the final output via weighted combination:

$$F(X) = W_{\text{clean}} \cdot E_{\text{clean}}(X) + W_{\text{adv}} \cdot E_{\text{adv}}(X) \tag{1}$$

However, experiments reveal significant limitations in the static fusion mechanism of SRAMoE. Figure 2 shows that the weight distribution of E_{clean} is smooth, favoring diverse global features, while the distribution is sharper of E_{adv}, focusing on local extrema to counter adversarial perturbations. This fundamental difference in feature extraction strategies stems from divergent optimization objectives: E_{clean} optimizes the standard classification loss L_{cls}, whereas E_{adv} incorporates an adversarial regularization term:

$$L_{\text{adv}}(X, Y) = L_{\text{cls}}(X, Y) + \lambda \cdot \text{AdvReg}(X, Y) \tag{2}$$

The adversarial regularization $\text{AdvReg}(X, Y)$ drives E_{adv} to focus on local feature perturbations, causing its weight updates to diverge from the optima of standard training. Due to the heterogeneity in weight distributions, merely adjusting weights via the router fails to coordinate E_{clean} and E_{adv}, limiting the balance between standard accuracy and robustness. See Section 4.2 for more results. The static limitations of SRAMoE compel us to seek a dynamic approach for resolution.

3.2 The Dynamic Design of DRAMoE

To address the fusion issues in SRAMoE, we further propose DRAMoE, which adopts a dynamic MoE architecture. Inspired by ADVMoE [38], DRAMoE refines expert granularity from model-level to channel-level, leveraging the MoE-CNN framework to enhance dynamic adaptability to input features. MoE-CNN uses routers to dynamically select channel-level experts in the CNN backbone network, where each expert processes a subset of the convolutional layer output channels for channel-level feature processing. However, ADVMoE has two limitations: its router relies on global average pooling mechanisms, struggles to capture channel-level feature importance variations, and underutilizes the potential of the router to guide AT. To address these, the proposed router-guided design leverages DCAR and RGAT, effectively enhancing the adversarial robustness of MoE-CNN.

Dynamic Channel Attention Router To address the global pooling limitation, we introduce DCAR, drawing on efficient attention mechanisms from MQA [20] and GQA [1] to improve router sensitivity to input features via a grouped self-attention mechanism, outperforming the static weight designs of conventional attention mechanisms like SENet [11] and CBAM [29], thereby enhancing perturbation resistance of routing within the MoE-CNN dynamic routing framework.

DCAR partitions input features $X \in \mathbb{R}^{B\times C\times H\times W}$ into K groups along the channel dimension, aligned with the number of MoE experts, each containing C/K channels, where $H \times W$ represents the spatial dimensions, and C denotes the number of feature channels. The k-th channel group is defined as:

$$X_k = X[:, (k-1)\cdot\frac{C}{K} : k\cdot\frac{C}{K}, :, :], \quad k = 1, 2, \ldots, K \tag{3}$$

Each group computes attention scores M_k and weights A_k using shared query Q_k, key K_k, and group-specific value V_k:

$$M_k = \frac{Q_k K_k^T}{\sqrt{\frac{C}{K}}}, \quad A_k = \mathrm{Softmax}(M_k \cdot \kappa) \tag{4}$$

Here κ is a learnable temperature coefficient controlling the sharpness of the attention distribution. The weighted output for each group is:

$$Z_k = V_k A_k^T \tag{5}$$

A dynamic routing module generates dual-path weights $W^{(k)} = [W_1^{(k)}, W_2^{(k)}]$, fusing attention outputs with original features:

$$Y_k = W_1^{(k)} \cdot Z_k + W_2^{(k)} \cdot X_k \tag{6}$$

The final feature $Y = \mathrm{Concat}(Y_1, Y_2, \ldots, Y_K)$ is obtained by concatenating group outputs. The channel of DCAR grouping collaborates with the MoE expert structure, where enhanced group features directly guide expert selection, improving allocation. Additionally, intra-group self-attention optimizes feature representation and sensitivity to adversarial perturbations, enabling the router to allocate expert resources more precisely and significantly enhancing perturbation sensitivity.

Router-Guided Adversarial Training Existing MoE-based AT methods underutilize the router to guide AT, to address this, we introduce RGAT, enhancing robustness by optimizing AEs generation and diversity regularization.

For AEs generation, traditional attacks apply blind perturbations across all channels, potentially wasting perturbation budgets in MoE-CNN architectures. Inspired by FAT [36], we employ a router-guided early-stopping mechanism (RG-ES) to focus on critical expert paths, avoiding overly aggressive AEs. The early-stopping is controlled by a tolerance parameter τ:

$$\tau^{(i+1)} = \begin{cases} \tau^{(i)} - 1, & \text{if } F(X_{\text{adv}}^{(i)}) \neq Y \text{ and } \tau^{(i)} > 0 \\ \tau^{(i)}, & \text{otherwise} \end{cases} \tag{7}$$

For diversity regularization, RGAT introduces a router-model consistency loss and a dual optimization mechanism. The consistency loss generates AEs that lead to differences between the model and the router output, following the minimax principle in AT:

$$\mathcal{L}_{\text{Adv}} = \frac{1}{N}\sum_{i=1}^{N}\left[D_{\text{KL}}\left(F(X_{\text{adv}}) \parallel F(X_{\text{clean}})\right) + \lambda(t)\cdot D_{\text{KL}}\left(G(X_{\text{adv}}) \parallel G(X_{\text{clean}})\right)\right] \tag{8}$$

Here, $\lambda(t)$ is a dynamic weight that balances the KL divergence term of the router, adjusted via a cosine annealing schedule. With the involvement of the router in AEs generation, perturbations are better focused on expert allocation.

During the AT phase, a dual optimization mechanism is adopted, first optimizing the model:

$$\mathcal{L}_{\text{Model}} = \frac{1}{N}\sum_{i=1}^{N}\left[\mathcal{L}_{\text{CE}}(F(X_{\text{clean}}), Y) + \beta\cdot D_{\text{KL}}(F(X_{\text{adv}}) \parallel F(X_{\text{clean}}))\right] \tag{9}$$

Here, β weights the KL divergence term to align the output of the model on clean examples and AEs.

This is followed by separately optimizing the router with expert assignment regularization:

$$\begin{aligned} \mathcal{L}_{\text{Router}} = \frac{1}{N}\sum_{i=1}^{N}\Big[& \gamma\cdot\mathcal{L}_{\text{CE}}(G(X_{\text{clean}}), Y) + \\ & \beta\cdot D_{\text{KL}}(G(X_{\text{adv}}) \parallel G(X_{\text{clean}})) + \\ & \delta\cdot(-H(G(X_{\text{clean}})) - H(G(X_{\text{adv}})))\Big] \end{aligned} \tag{10}$$

$$H(p) = -\sum_{j=1}^{n} p_j \log(p_j), \quad p = \mathrm{softmax}(G(X)) \tag{11}$$

Here, n is the number of experts, γ weights the classification loss of the router, while δ encourages uniform expert assignments by maximizing the entropy $H(p)$ of the output distribution of the router, mitigating assignment degradation in AT.

4 Experiments

4.1 Experimental Setup

Datasets and model setup: We conduct experiments on CIFAR-10, CIFAR-100 [13], and Tiny ImageNet [15], three standard image classification datasets with 10, 100, and 200 classes, respectively. To evaluate model performance across these datasets, we utilize three backbone networks: ResNet-18[10], ResNet-34, and Wide-ResNet-28-10(WRN-28-10) [33], representing residual and wide residual architectures.

Evaluation Metrics: We assess defense performance using two metrics: Standard Accuracy (*Std*), which measures performance on clean examples, and Robust Accuracy (*Adv*), which evaluates defense against AEs, tested with PGD-generated [18] AEs by default.

Experimental Details: During training, we employ RGAT, dynamically adjusting AE generation intensity with the hyperparameter τ, which varies across epochs: $\tau = 0$ for the first 20% of epochs, $\tau = 1$ for 20%–40%, and $\tau = 2$ for 40%–60%. Other RGAT hyperparameters include $\lambda(t)$ with cosine annealing ($\lambda_{\max} = 0.001$), $\beta = 3.0$, $\gamma = 6.0$, and $\delta = 0.001$. Attack parameters are set uniformly: perturbation magnitude $\epsilon = 8/255$, step size $\alpha = 2/255$, and iteration steps $i = 10$. MoE settings include a model scale of $r = 0.5$ with $n = 2$ experts.

MoE-based AT baselines setup: Our method leverages a novel network architecture. For a comprehensive evaluation, we compare the following baseline models, covering both MoE-based dynamic routing methods and non-MoE static sparse methods:

1. **PGD-AT (Sparse) [19]**: A static sparse subnetwork generated from a dense CNN via structured pruning, trained with two-stage AT (100 epochs of mask optimization + 100 epochs of weight fine-tuning). Its fixed paths contrast with the dynamic routing of MoE-CNN.

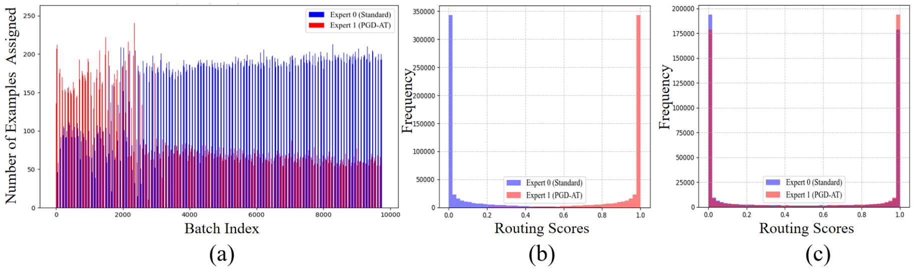
2. **PGD-AT (MoE-CNN) [38]**: Single-stage AT on MoE-CNN, jointly optimizing router and expert weights, with $n = 2$ experts and a model scale of $r = 0.5$.

3. **ADVMoE [38]**: Alternating AT on MoE-CNN using a dual-layer optimization strategy, optimizing router and expert weights alternately, with $n = 2$ experts and a model scale of $r = 0.5$.

For a fair comparison, the weight retention ratio of PGD-AT (Sparse) aligns with the model scale parameter r of MoE-CNN. All models use TRADES as the default training strategy.

4.2 Limitations of SRAMoE

SRAMoE aims to fuse the advantages of pre-trained models through MoE and is expected to leverage the allocation of routers to standard or adversarial experts based on the characteristics of the input data. However, the experimental results reveal the limitations of the routers: Figure 3 (a) shows that its allocation behavior tends to E_{clean} (Expert 0), failing to fully utilize the potential of E_{adv} (Expert 1). Figure 3 (b) and (c) further reveal the differences in the distribution of router weights in different types of examples. This deviation is attributed to the difficulty of routers in distinguishing the characteristics of clean examples from AEs, resulting in inefficient allocation. The static routing mechanisms have difficulty in effectively coordinating experts with different optimization objectives, limiting their robustness and generalization capabilities, and motivating a dynamic approach.



(a) (b) (c)

Fig. 3. Routing analysis of SRAMoE: (a) shows the allocation of experts for each batch; (b) and (c) represent the routing score distributions for clean examples and AEs of different experts.

4.3 Performance Evaluation

To evaluate the performance of DRAMoE, we evaluate DRAMoE against baselines PGD-AT (Sparse), PGD-AT (MoE-CNN), and ADVMoE on different datasets and model architectures, with results detailed in Table 1. DRAMoE demonstrates superior performance in both clean examples and AEs. For instance, on CIFAR-10 with the ResNet-18 architecture, DRAMoE achieves a robust accuracy of 62.13%, surpassing ADVMoE by 4.02%, and a standard accuracy of 84.86%, exceeding ADVMoE by 2.82%. In contrast, PGD-AT (Sparse) is hindered by its inability to adapt to heterogeneous data distributions, while PGD-AT (MoE-CNN) suffers from inefficient router-expert coordination. Similarly, ADVMoE's global pooling-based router overlooks fine-grained adversarial features and underutilizes the router to guide AT, limiting its robustness. By leveraging the channel-level attention of DCAR for precise feature capture and the

Table 1. Comparison of defense performance between our method and existing MoE-based AT baselines.

		CIFAR-10		CIFAR-100		Tiny ImageNet	
Architecture	Method	*Std*	*Adv*	*Std*	*Adv*	*Std*	*Adv*
ResNet-18	PGD-AT (Sparse)	80.63	53.93	57.77	29.33	50.86	37.36
	PGD-AT (MoE)	78.24	52.66	53.68	25.67	47.68	35.77
	ADVMoE	82.04	58.11	54.58	31.74	53.44	40.93
	Ours	**84.86**	**62.13**	**57.01**	**32.59**	**53.71**	**41.23**
ResNet-34	PGD-AT (Sparse)	81.34	56.23	58.21	29.66	53.63	38.56
	PGD-AT (MoE)	78.96	54.87	54.77	26.12	48.48	36.42
	ADVMoE	**84.27**	63.54	56.57	31.98	53.21	41.27
	Ours	84.03	**64.53**	**58.67**	**33.42**	**54.53**	**42.58**
WRN-28-10	PGD-AT (Sparse)	84.47	57.95	60.39	31.83	54.39	40.26
	PGD-AT (MoE)	79.56	55.73	56.39	28.94	49.97	37.61
	ADVMoE	88.26	62.75	60.91	35.75	55.23	42.17
	Ours	**88.41**	**65.09**	**61.85**	**37.41**	**55.76**	**43.09**

diversity regularization of RGAT for optimized router decisions, DRAMoE significantly enhances expert allocation efficiency for both clean examples and AEs, achieving a superior balance between standard accuracy and robustness.

To further validate the effectiveness of DRAMoE, we compare it with static network AT strategies, including PGD-AT [18], TRADES [35], MART [28], and FAT [36], as shown in Table 2. Results demonstrate that DRAMoE, leveraging its dynamic network architecture, significantly outperforms static network methods in robustness. On the CIFAR-10 dataset, DRAMoE achieves a robust accuracy improvement of approximately 5.48% over FAT under the ResNet-18 architecture, while maintaining competitive standard accuracy, with particularly notable performance on the WRN-28-10 architecture. These findings not only confirm the superiority of DRAMoE in AT but also highlight the potential of the MoE architecture in dynamically adapting to heterogeneous data distributions.

4.4 Ablation Evaluation

To further validate the effectiveness of our method, we perform various ablation studies from multiple perspectives.

First, we evaluate the impact of DCAR and RGAT on model performance, as shown in Table 3. Using only the MoE framework, the model is limited by distribution bias between clean examples and AEs, degrading performance. Adding DCAR enables the router to capture local adversarial features via channel-level attention, improving standard accuracy, though robust accuracy slightly decreases. Incorporating RGAT further enhances robust accuracy through effective router guidance. Together, DCAR and RGAT achieve an optimal balance between standard accuracy and robust accuracy.

Table 2. Comparison of defense performance among our method and static network AT strategies.

		CIFAR-10		CIFAR-100	
Architecture	Method	*Std*	*Adv*	*Std*	*Adv*
ResNet-18	MART	81.65	53.75	52.91	30.87
	TRADES	81.82	52.48	55.27	27.68
	PGD-AT	83.10	52.05	56.88	28.33
	FAT	**85.47**	56.65	56.94	30.69
	Ours	84.86	**62.13**	**57.01**	**32.59**
ResNet-34	MART	81.15	55.83	53.81	31.85
	TRADES	83.74	53.12	55.97	28.69
	PGD-AT	85.11	54.53	58.37	28.74
	FAT	**86.37**	56.35	58.17	30.55
	Ours	84.03	**64.53**	**58.67**	**33.42**
WRN-28-10	MART	81.27	57.69	56.68	32.02
	TRADES	85.22	56.81	56.12	29.74
	PGD-AT	86.21	56.02	**62.04**	29.51
	FAT	87.75	59.67	60.87	32.82
	Ours	**88.41**	**65.09**	61.85	**37.41**

Table 3. Ablation study evaluating the effects of MoE, DCAR, and RGAT components on model performance on ResNet-18 for CIFAR-10.

MoE	DCAR	RGAT	*Std*	*Adv*
✓	-	-	79.10	53.07
✓	✓	-	82.99	59.29
✓	-	✓	81.49	64.75
✓	✓	✓	**84.86**	**62.13**

Second, we compare three attention mechanisms: Global Self-Attention (GSA), Channel-Grouped Self-Attention (CGSA), and DCAR, as shown in Table 4. GSA captures global features but lacks sensitivity to channel-level adversarial perturbations; CGSA improves local feature perception through grouping but is less efficient; DCAR, with shared transformations and dynamic routing weights, optimizes feature representation and perturbation sensitivity, outperforming GSA.

Finally, we compare two RGAT early-stopping strategies: Early-Stopping without Router Guidance (NoRG-ES) and RG-ES. On CIFAR-10, RG-ES leverages router consistency loss to improve standard accuracy and robust accuracy, demonstrating the enhanced adaptability of RGAT to heterogeneous data distributions.

Table 4. Performance comparison of DRAMoE variants with different attention mechanisms and early-stopping strategies on ResNet-18 for CIFAR-10.

Method	*Std*	*Adv*
Ours (GSA)	83.44	61.93
Ours (CGSA)	85.46	60.19
Ours (DCAR)	**84.86**	**62.13**
Ours (NoRG-ES)	83.54	60.45
Ours (RG-ES)	**84.86**	**62.13**

4.5 Visualization Evaluation

To investigate the feature representation and robustness of DRAMoE, we analyze its behavior using attention heatmaps (the right of Figure 4) and t-SNE [17] feature clustering (the left of Figure 4). Heatmaps show that a standard model without AT focuses on target objects for clean examples but exhibits significant attention shifts under AEs, whereas DRAMoE maintains stable attention in both scenarios. T-SNE clustering reveals that the standard model produces dispersed feature clusters for AEs, indicating limited discriminative capability, while DRAMoE generates compact clusters for both clean examples and AEs, confirming its ability to effectively capture data distributions and demonstrating its robustness.

To test the performance of DRAMoE in other attack scenarios, we test it on other white-box attacks using WRN-28-10, as shown in Figure 5. DRAMoE consistently demonstrates stable robustness, particularly against strong attacks like AutoAttack, highlighting its generalization and defensive capability in complex adversarial settings.

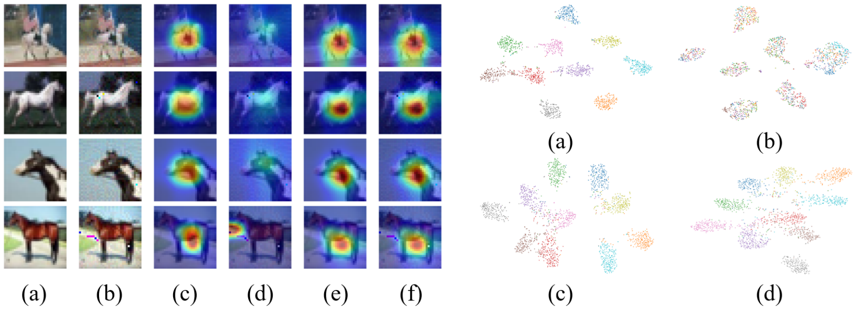


Fig. 4. Class activation mapping heatmaps and clustering diagrams of different models on clean examples and AEs. The left figure (a) shows clean examples, (b) shows AEs, (c-d) and (e-f) show the heatmaps of the standard model and DRAMoE on clean examples and AEs. The right figures (a-b) and (c-d) show the clustering effects of the standard model and DRAMoE on clean examples and AEs, respectively.

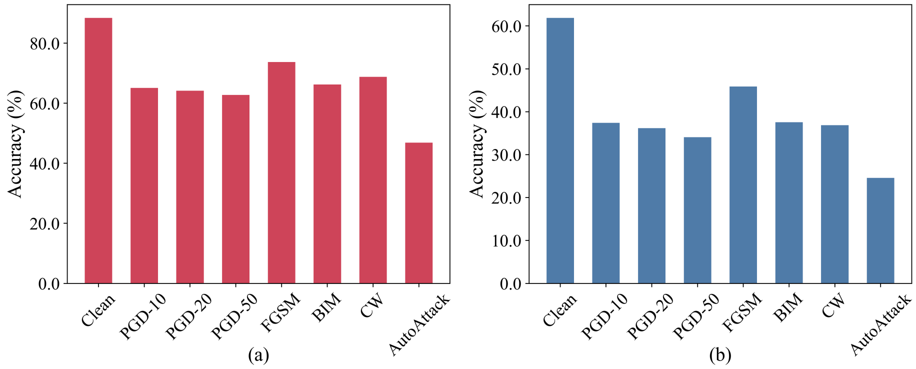


Fig. 5. Standard Accuracy and Robust Accuracy on (a) CIFAR-10 and (b) CIFAR-100 under PGD (10/20/50 steps), FGSM, BIM, CW, and AutoAttack.

5 Conclusion

AT enhances the robustness of DL models but often compromises the accuracy of clean examples, making the trade-off between standard accuracy and robustness a key research challenge. To address this, we propose DRAMoE, a novel AT strategy that integrates MoE with AT. Employing a divide-and-conquer approach, DRAMoE leverages DCAR and RGAT to enhance the dynamic sensitivity of the router to adversarial features and guide AT. This improves expert allocation specificity, effectively balancing downstreaming standard accuracy and robustness.

Extensive experiments validate the effectiveness of DRAMoE in enhancing both standard accuracy and robustness, opening new avenues for AT research. Meanwhile, the rapid advancement of large language models presents unprecedented opportunities for improving adversarial robustness in DL. We are actively exploring the role of large language models across various components and architectures, seeking innovative strategies to further advance this field.

References

1. Ainslie, J., Lee-Thorp, J., De Jong, M., Zemlyanskiy, Y., Lebrón, F., Sanghai, S.: Gqa: Training generalized multi-query transformer models from multi-head checkpoints. arXiv preprint arXiv:2305.13245 (2023)
2. Athalye, A., Carlini, N., Wagner, D.: Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In: International conference on machine learning. pp. 274–283. PMLR (2018)
3. Breiman, L.: Bagging predictors. Machine learning **24**, 123–140 (1996)
4. Cai, W., Jiang, J., Wang, F., Tang, J., Kim, S., Huang, J.: A survey on mixture of experts. arXiv preprint arXiv:2407.06204 (2024)

5. Croce, F., Hein, M.: Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: International conference on machine learning. pp. 2206–2216. PMLR (2020)
6. Deng, B., Zhang, D., Dong, F., Zhang, J., Shafiq, M., Gu, Z.: Rust-style patch: A physical and naturalistic camouflage attacks on object detector for remote sensing images. Remote. Sens. **15**, 885 (2023), https://api.semanticscholar.org/CorpusID:256657632
7. Duan, R., Chen, Y., Niu, D., Yang, Y., Qin, A.K., He, Y.: Advdrop: Adversarial attack to dnns by dropping information. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 7506–7515 (2021)
8. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences **55**(1), 119–139 (1997)
9. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
11. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7132–7141 (2018)
12. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E.: Adaptive mixtures of local experts. Neural computation **3**(1), 79–87 (1991)
13. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
14. Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial machine learning at scale. arXiv preprint arXiv:1611.01236 (2016)
15. Le, Y., Yang, X.: Tiny imagenet visual recognition challenge. CS 231N **7**(7), 3 (2015)
16. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436–444 (2015)
17. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)
18. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017)
19. Sehwag, V., Wang, S., Mittal, P., Jana, S.: Hydra: Pruning adversarially robust neural networks. Advances in Neural Information Processing Systems **33**, 19655–19666 (2020)
20. Shazeer, N.: Fast transformer decoding: One write-head is all you need, 2019. URL https://arxiv. org/abs (1911)
21. Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., Dean, J.: Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. arXiv preprint arXiv:1701.06538 (2017)
22. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013)
23. Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., Madry, A.: There is no free lunch in adversarial robustness (but there are unexpected benefits). arXiv preprint arXiv:1805.12152 **2**(3) (2018)
24. Wang, X., Yu, F., Dunlap, L., Ma, Y.A., Wang, R., Mirhoseini, A., Darrell, T., Gonzalez, J.E.: Deep mixture of experts via shallow embedding. In: Uncertainty in artificial intelligence. pp. 552–562. PMLR (2020)

25. Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., Gu, Q.: Improving adversarial robustness requires revisiting misclassified examples. In: International Conference on Learning Representations (2020), https://api.semanticscholar.org/CorpusID:211548864
26. Wei, X., Zhao, S., Li, B.: Revisiting the trade-off between accuracy and robustness via weight distribution of filters. IEEE Transactions on Pattern Analysis and Machine Intelligence (2024)
27. Wolpert, D.H.: Stacked generalization. Neural networks **5**(2), 241–259 (1992)
28. Wong, E., Rice, L., Kolter, J.Z.: Fast is better than free: Revisiting adversarial training. arXiv preprint arXiv:2001.03994 (2020)
29. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: Convolutional block attention module. In: Proceedings of the European conference on computer vision (ECCV). pp. 3–19 (2018)
30. Xie, C., Tan, M., Gong, B., Wang, J., Yuille, A.L., Le, Q.V.: Adversarial examples improve image recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 819–828 (2020)
31. Xu, W., Evans, D., Qi, Y.: Feature squeezing: Detecting adversarial examples in deep neural networks. arXiv preprint arXiv:1704.01155 (2017)
32. Yang, B., Bender, G., Le, Q.V., Ngiam, J.: Condconv: Conditionally parameterized convolutions for efficient inference. Advances in neural information processing systems **32** (2019)
33. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv preprint arXiv:1605.07146 (2016)
34. Zhang, D., Liang, Y., Huang, Q., Huang, X., Liao, P., Yang, M., Zeng, L.: Freqat: An adversarial training based on adaptive frequency-domain transform. In: International Conference on Advanced Data Mining and Applications. pp. 287–301. Springer (2024)
35. Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., Jordan, M.: Theoretically principled trade-off between robustness and accuracy. In: International conference on machine learning. pp. 7472–7482. PMLR (2019)
36. Zhang, J., Xu, X., Han, B., Niu, G., Cui, L., Sugiyama, M., Kankanhalli, M.: Attacks which do not kill training make adversarial learning stronger. In: International conference on machine learning. pp. 11278–11287. PMLR (2020)
37. Zhang, L., Huang, S., Liu, W.: Enhancing mixture-of-experts by leveraging attention for fine-grained recognition. IEEE Transactions on Multimedia **24**, 4409–4421 (2022). https://doi.org/10.1109/TMM.2021.3117064
38. Zhang, Y., Cai, R., Chen, T., Zhang, G., Zhang, H., Chen, P.Y., Chang, S., Wang, Z., Liu, S.: Robust mixture-of-expert training for convolutional neural networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 90–101 (2023)