RoME: Robust Mixture of Low-Rank Experts against Multiple Adversarial Perturbations

Woo Jae Kim[1], Kyle Min[2], Suhyeon Ha[1], Joonsung Jeon[1], and Sung-eui Yoon[1]

[1] KAIST, Daejeon, Korea
[2] Oracle, Seattle WA, USA
{wkim97, suhyeon.ha, mikeraph}@kaist.ac.kr, sungeui@kaist.edu
kyle.min@oracle.com

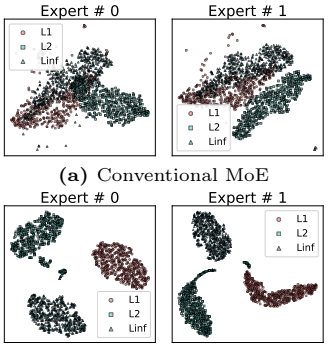
Abstract. Multi-perturbation adversarial training (MAT) aims to achieve robustness against multiple ℓ_p perturbations but suffers from robustness trade-offs between different threats. To address this, we employ a mixture of experts (MoE) to route different threats through distinct model pathways. However, naïve application of MoE encounters two critical challenges: experts tend to overlook threat-specific features and redundantly capture features shared across threats, and gating networks suffer from *threat-agnostic routing* where they learn nearly identical routing patterns across threats, thus preventing the construction of threat-specific model pathways. To this end, we propose Robust Mixture of Low-Rank Experts (RoME), where each expert is a low-rank additive update to the shared backbone, allowing it to capture threat-common features while experts focus on threat-specific information. To address threat-agnostic routing, RoME introduces (i) dual-scale gating that exploits threat-discriminative signals from local and global level features, and (ii) threat-guided gating diversification that enforces diverse expert utilization across threats. Extensive experiments demonstrate that RoME outperforms existing state-of-the-art MAT in union robustness and natural accuracy and improves robustness against unseen threats. Codes are available at `https://github.com/wkim97/RoME`.

Keywords: Multi-Perturbation Adversarial Training · Mixture of Experts · Adversarial Robustness

1 Introduction

Adversarial training [21,43] has become one of the most effective defenses against adversarial threats. However, traditional adversarial training typically focuses on a single type of threat (*e.g.*, ℓ_∞ perturbations), remaining vulnerable to unseen threats during inference [44, 59]. Multi-perturbation adversarial training (MAT) [9, 28, 42, 44, 59] addresses this by training on diverse ℓ_p threats simultaneously, improving robustness across multiple threats.

Despite its effectiveness, MAT faces robustness trade-offs when learning on diverse threats, suffering from suboptimal robustness on individual threat types [9,



(a) Conventional MoE

(b) Ours (Low-rank experts)

Fig. 1: t-SNE visualization of expert outputs. (a) In conventional MoE, disjoint FFN experts redundantly capture similar **threat-common features**, while (b) our low-rank experts effectively capture **threat-specific features**.

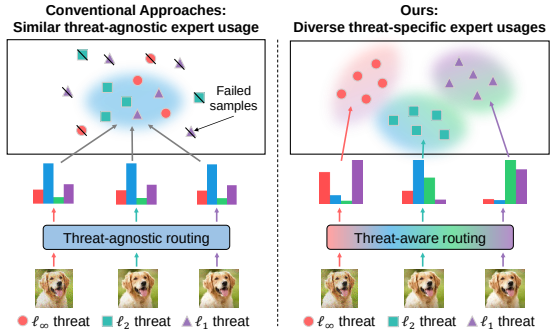


Fig. 2: Conceptual visualization of conventional mixture of experts and our approach. Conventional methods (left) face the **threat-agnostic routing** issue and route different threats through similar expert combinations, resulting in a single shared model pathway for distinct threats. Our approach (right) effectively routes each threat through distinct expert combinations, constructing multiple threat-specific model pathways.

28, 59]. This degradation arises because different adversarial threats induce distinct distributional shifts, creating conflicting optimization objectives during training [9,28]. Existing approaches [9,28,59] force these distinct threats through a single, fixed architectural pathway, where feature representations learned to be robust against one threat may fail to transfer robustness to others.

To address this, we employ mixture of experts (MoE) [11, 18, 56, 67] to route different threats through distinct model pathways, each learning representations robust to a different type of threat. However, naïve application of MoE [18,47,56] encounters two critical challenges. First, conventional experts, typically implemented as multiple independent FFN layers, struggle to capture features unique to each threat. Adversarial examples crafted from an image naturally share the same underlying image content [26, 78], and certain threats even share overlapping features [9] (*e.g.*, ℓ_1 and ℓ_∞ robustness partially transfer to ℓ_2). Without a shared backbone to separate threat-common from threat-specific features [69], conventional experts redundantly capture the former as shown in Fig. 1a.

The shared features among threats further raise a second challenge of *threat-agnostic routing*, where the gating network routes different threats to similar expert combinations as shown in Fig. 2, failing to construct distinct pathways for each threat. This occurs because the overlapping features across threat types provide insufficient discriminative signal, preventing the gating network from distinguishing one threat from another. This forces a single model pathway to learn robust representations against multiple threats, facing the same multi-threat trade-off in existing MAT methods.

To address these challenges, our key insight is two-fold. First, inspired by a recent approach [69] in multi-task learning that handles different tasks using low-

rank experts on top of a shared backbone, we shift from conventional disjoint experts and implement each expert as a **low-rank additive update** to the shared backbone weights. This allows the backbone to capture threat-common features, while each expert focuses on learning threat-specific information.

To address threat-agnostic routing, we make an observation that different threats exhibit discriminative cues at different levels of granularity (Fig. 5 in Sec. 4.3); ℓ_1 threats are more distinguishable at local patch-level features due to their sparse perturbations, while ℓ_∞ threats show clearer discrimination at global image-level features due to their uniform perturbations across the image. We thus propose **threat-distinguishing dual-scale gating** that combines both scales to provide richer discriminative signals to gating networks. However, even with richer input signals, the gating network may still converge to threat-agnostic expert assignments without explicit routing supervision. We thus further introduce **threat-guided gating diversification**, which supervises gating networks to enforce diverse routing patterns across threats.

Building on these insights, we propose Robust Mixture of Low-Rank Experts (RoME) consisting of: (i) low-rank experts (Sec. 4.2) for learning threat-specific model pathways, (ii) threat-distinguishing dual-scale gating (Sec. 4.3) for discriminative gating signals, and (iii) threat-guided gating diversification (Sec. 4.4) for diverse expert utilization across threats. These components effectively route each threat to distinct expert combinations (Fig. 2), with each expert learning distinct threat-specific features (Fig. 1b), enabling threat-adaptive model pathways that mitigate cross-threat trade-offs.

We conduct extensive experiments on CIFAR-10, ImageNet-100 and ImageNet-1K, demonstrating that RoME outperforms existing state-of-the-art MAT methods in union robustness and natural accuracy (Sec. 5.2). Diverse representations captured with multiple model pathways also improve robustness against threats unseen during training (Sec. 5.2). Ablation studies (Sec. 5.3) validate the importance of each component of RoME, and analysis (Sec. 5.4) shows that our framework enables threat-specific representations for improved performance.

In summary, our contributions are as follows:

- We identify threat-agnostic routing problem when applying MoE to MAT, where the gating network learns similar expert routing for different threats.
- To this end, we introduce RoME, a novel mixture of low-rank experts framework for adversarial training on multiple threat types via threat-distinguishing dual-scale gating and threat-guided gating diversification.
- Extensive experiments demonstrate that our approach outperforms existing state-of-the-art MAT in multi-threat robustness across multiple benchmarks.

2 Related Works

2.1 Multi-Perturbation Adversarial Training (MAT)

Adversarial training, which minimizes the worst-case loss over adversarial examples [21, 43], has become the de facto adversarial defense. Subsequent work

improved trade-off between robustness and natural accuracy [60, 72], enhanced training efficiency [55, 63], and improved scalability to large-scale datasets [57, 61, 66]. Despite these advances, most methods optimize for a single adversarial threat (*e.g.*, ℓ_∞-bounded), limiting robustness against different threats.

To improve robustness across diverse adversarial threats, early work [59] proposed training a model on multiple ℓ_p threats. Subsequent works improved robustness or training efficiency via worst-case descent directions [44], stochastic adversarial sampling [42], alternation between extreme norms [9], parameter-space interpolation [10], logit-pairing with gradient projection [28], or logit-space regularization [12]. While effective, these methods learn robustness against distinct threats within a single model representation space, where optimizing for one threat can interfere with others [9,28,59] and degrade their robustness. MORE [5] employs mixture-of-experts but relies on gating without explicit threat-aware guidance, which can face difficulty distinguishing between threats. In contrast, our RoME learns diverse model pathways each tailored to different threats for mitigating cross-threat robustness trade-off.

2.2 Mixture of Experts

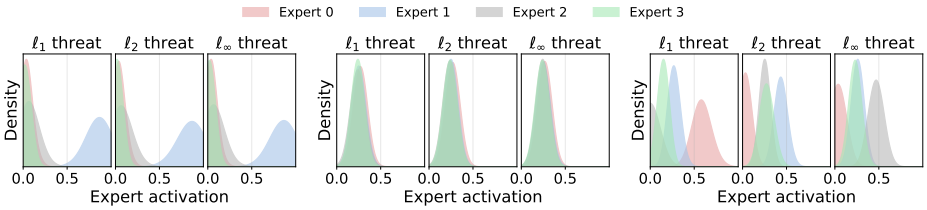
Mixture of Experts (MoE) [27, 56] learns multiple experts and a gating network that routes each input to a subset of experts, enabling input-adaptive model specialization. In Transformers, this is typically implemented by replacing feed-forward networks (FFNs) with multiple independent FFN experts [18, 47, 56]. Different variants have enhanced its routing stability [11,18], scalability [34], and expert selection [51, 77]. MoE has been applied across multi-task learning [3, 40, 69], continual learning [35, 70], and large-scale vision models [52]. More recently, MoE has been applied with parameter-efficient adapters [16, 19, 64, 68]. While parallel lines of works explored adversarial robustness of MoE architectures [48, 73] or used MoE for improving robustness against a single threat [45, 49, 50, 74], our work addresses an orthogonal problem of resolving cross-threat robustness trade-offs when training on multiple distinct threat types. To this end, we tackle the redundant features and threat-agnostic routing issues of conventional MoE through low-rank experts, dual-scale gating, and gating diversification.

3 Preliminary

We consider a classification model $f_\theta : \mathcal{X} \to \mathcal{Y}$ parameterized by θ, where $\mathcal{X}$ and $\mathcal{Y}$ denote the input and label spaces, and $(x, y) \sim \mathcal{D}$ is a sample from data distribution $\mathcal{D}$. Given adversarial threats $\{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_\infty\}$ with perturbation budgets $\{\epsilon_1, \epsilon_2, \epsilon_\infty\}$, multi-perturbation adversarial training (MAT) aims to solve:

$$\min_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}}\, \phi(\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_\infty), \tag{1}$$

where $\mathcal{L}_p = \max_{\delta_p \in \mathcal{A}_p(x,\epsilon_p)} \mathcal{L}(f_\theta(x + \delta_p), y)$ is adversarial loss [43] under threat $\mathcal{A}_p$ for $p \in \{1, 2, \infty\}$ [9, 28, 59], and $\phi : \mathbb{R}^3 \to \mathbb{R}$ aggregates losses across threats.



(a) Naïve approach **(b)** Load balancing [56] **(c)** Ours (RoME)

Fig. 3: Gaussian KDE analysis on expert activations across different threat types on CIFAR-10. (a) Naïve MoE suffers from routing collapse (Expert 1 dominates) and threat-agnostic routing (similar routing across threats). (b) Load balancing [56] resolves routing collapse but exhibits threat-agnostic routing. (c) Our RoME addresses threat-agnostic routing issue and learns diverse, threat-specific expert combinations.

MAX [59] implements aggregation as $\phi_{\text{MAX}}(\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_\infty) = \max\{\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_\infty\}$ to optimize on the worst-case threat at each iteration. **RANDOM** [42] implements aggregation as $\phi_{\text{RANDOM}}(\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_\infty) = \mathcal{L}_p$, where a threat $p \sim \text{Cat}(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ is sampled stochastically.

4 RoME

4.1 Limitations of Conventional MoE

We employ mixture of experts (MoE) to construct multiple threat-specific model pathways for improved multi-threat robustness. However, unlike prior MoE applications with clearly distinguishable inputs (*e.g.*, different tasks or languages) [3, 4,17,75,76], adversarial threats share the identical semantic content of the underlying image being perturbed, which creates a strong inductive bias [26,78]. Without an architectural component to capture these common characteristics [69], experts are prone to capturing redundant similar representations as shown in Fig. 1, thus preventing the construction of threat-specific model pathways.

These threat-common features further induce *threat-agnostic routing* problem, where the gating network receives insufficient discriminative signal and routes different threats to similar expert combinations, failing to mitigate cross-threat trade-offs. To demonstrate this issue, we analyze the gating weight distributions across threats in Fig. 3. We train ViT-B [15] on CIFAR-10 [30] with 4 FFN experts [47, 56] for each Transformer block using RANDOM [42] (full details in Sec. A1 of supplementary) and visualize token-averaged gating weights from the final layer over 10K test samples.

We first consider a naïve approach that trains the MoE with adversarial loss only (Eq. 1). As shown in Fig. 3a, this approach faces (i) *routing collapse*, where only a few experts (Expert 1) dominate, and (ii) *threat-agnostic routing*, where all threats exhibit nearly identical gating distributions. To address routing collapse, we apply load balancing loss [56,64] widely used in conventional MoE [11,56,67] to encourage uniform expert utilization, with its loss coefficient set to 0.5 following existing protocol [64]. As shown in Fig. 3b, while load balancing resolves

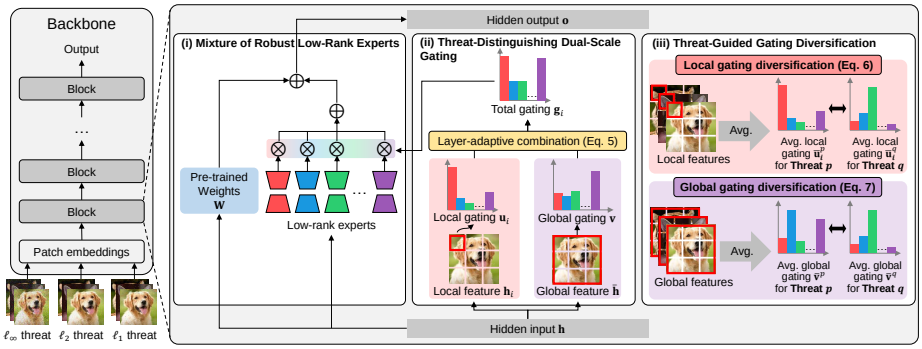


Fig. 4: Overview of RoME. (i) We train a mixture of robust low-rank experts, where a shared backbone captures threat-common features while low-rank experts focus on threat-specific information. To address threat-agnostic routing, where the gating assigns similar expert combinations across threats, we propose (ii) threat-distinguishing dual-scale gating, which leverages local patch-level and global image-level features to better distinguish between threats, and (iii) threat-guided gating diversification, which ensures different threats are routed to distinct expert combinations.

routing collapse, it still suffers from threat-agnostic routing, where the gating distributions remain similar across threats. Evaluation across different loss coefficients in Fig. A1 also consistently exhibit threat-agnostic routing problem. This reveals that existing MoE techniques alone are insufficient, motivating our framework to learn distinct, threat-aware expert combinations (Fig. 3c).

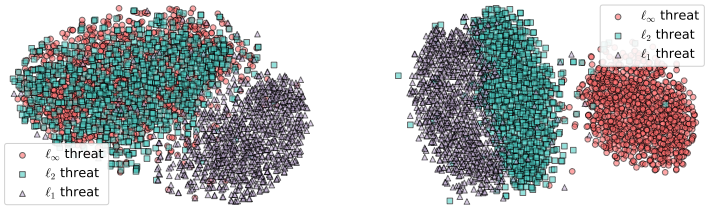
In this work, we propose RoME (Robust Mixture of Low-Rank Experts), illustrated in Fig. 4, consisting of three components: (i) low-rank experts (Sec. 4.2) to capture threat-specific features, (ii) threat-distinguishing dual-scale gating (Sec. 4.3) to provide gating with threat-discriminative signals, and (iii) threat-guided gating diversification (Sec. 4.4) to enforce routing diversity across threats.

4.2 Mixture of Robust Low-Rank Experts

To prevent experts from redundantly learning shared information common across threats, we implement each expert as a low-rank additive update to the backbone weights. We allow the shared backbone to capture threat-common features, while the low-rank experts focus on capturing threat-specific information.

Inspired by its recent success, we implement these low-rank experts as a LoRA [24] module. Each expert $k \in \{1, \dots, K\}$ learns an additive adjustment to the backbone layer with weight matrix $\mathbf{W} \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ via low-rank matrices $\mathbf{B}_k \in \mathbb{R}^{d_{\text{out}} \times r}$ and $\mathbf{A}_k \in \mathbb{R}^{r \times d_{\text{in}}}$, where $r \ll d_{\text{out}}, d_{\text{in}}$. Given input feature $\mathbf{h}_i \in \mathbb{R}^{d_{\text{in}}}$ at token i, the output $\mathbf{o}_i \in \mathbb{R}^{d_{\text{out}}}$ is computed as:

$$\mathbf{o}_i = \mathbf{W}\mathbf{h}_i + \sum_{k=1}^{K} g_{i,k} \cdot \mathbf{B}_k \mathbf{A}_k \mathbf{h}_i. \tag{2}$$



(a) Patch-level features **(b)** Image-level features

Fig. 5: t-SNE visualization of threat separability at different feature levels. (a) Local patch-level features better separate ℓ_1 threats (silhouette: 0.394) but show overlap for ℓ_∞ and ℓ_2. (b) Global image-level features show the complementary pattern, better distinguishing ℓ_∞ (silhouette: 0.611) over ℓ_1 and ℓ_2.

Gating weight $\mathbf{g}_i = [g_{i,1}, \ldots, g_{i,K}]^\top \in \mathbb{R}^K$ is predicted for each token i (*i.e.*, patch) in an image by a gating network such that $g_{i,k} \in [0, 1]$ and $\sum_{k=1}^{K} g_{i,k} = 1$.

4.3 Threat-Distinguishing Dual-Scale Gating

To address threat-agnostic routing, we propose dual-scale gating that leverages both local patch-level and global image-level features to distinguish between threats. This is based on the intuition that adversarial threats exhibit discriminative patterns at different feature levels. For instance, ℓ_1 threats create sparse perturbations on specific pixels, best captured by patch-level features, while ℓ_∞ perturbations are spatially uniform and are more evident in image-level features.

To verify this, we visualize feature separability using t-SNE [41] (Fig. 5) from ViT-B [15] trained with RANDOM strategy [42] on CIFAR-10. For patch-level features (Fig. 5a), we compute separability for all T token positions through silhouette coefficient score [53]. For image-level features (Fig. 5b), we average features across all patches and compute separability score. Our analysis reveals an interesting observation that local features show substantially higher separability for ℓ_1 threats (mean silhouette score: 0.394 ± 0.155) compared to ℓ_∞ (0.125 ± 0.077) and ℓ_2 (0.078 ± 0.053). Conversely, global image-level features exhibit the opposite trend (ℓ_1: 0.303 , ℓ_2: 0.281, ℓ_∞: 0.611), indicating that ℓ_∞ threats are more distinguishable compared to ℓ_1 and ℓ_2 threats. Analysis across layers and patches in Fig. A4 verifies the same phenomenon.

Based on these observations, we design a dual-scale gating comprised of local gating and global gating, each capturing patch-wise and image-level perturbation patterns, respectively. Given the hidden state $\mathbf{h}_i \in \mathbb{R}^{d_{\text{in}}}$ at token i, the local gating network $\text{MLP}_{\text{local}}$ predicts patch-wise gating weight $\mathbf{u}_i$ as follows:

$$\mathbf{u}_i = \text{softmax}\left(\text{MLP}_{\text{local}}(\mathbf{h}_i)\right) \in \mathbb{R}^K. \tag{3}$$

For image-level information, we average hidden states across all T tokens to obtain $\bar{\mathbf{h}} = \frac{1}{T}\sum_{i=1}^{T} \mathbf{h}_i$ and pass it through the global gating network $\text{MLP}_{\text{global}}$:

$$\mathbf{v} = \text{softmax}\left(\text{MLP}_{\text{global}}(\bar{\mathbf{h}})\right) \in \mathbb{R}^K. \tag{4}$$

This outputs a single gating weight vector $\mathbf{v}$ for an image.

Layer-adaptive combination. Early Transformer layers encode token-level information while deeper layers capture global semantic-level features [20, 37], creating a layer-wise bias where global features are under-represented in early layers and local features in deeper layers. Since both features are crucial for threat discrimination (Fig. 5), we compensate for this bias as:

$$\mathbf{g}_i = (1 - \beta(l)) \cdot \mathbf{u}_i + \beta(l) \cdot \mathbf{v}, \tag{5}$$

where $\mathbf{g}_i = [g_{1,i}, \ldots, g_{K,i}]^T$ is the final gating weight for token i, and $g_{k,i}$ denotes the weight for expert k. The layer-adaptive coefficient $\beta(l) = \sigma\left(s \cdot \left(1 - \frac{l}{L}\right) - b\right) \in [0, 1]$ emphasizes global gating in early layers and local gating in deeper layers, with $\sigma(\cdot)$ the sigmoid function, L the total number of layers, and s, b controlling transition across layers [37]. Since the structural property of Transformer layer hierarchy is well-established [20,37], we encode it directly as a fixed prior, which empirically outperforms the learnable variant (Table 4 in Sec. 5.3).

4.4 Threat-Guided Gating Diversification

While our dual-scale gating captures discriminative cues for each threat, it still requires explicit supervision to learn diverse, threat-specific expert combinations. To this end, we introduce a regularization loss that encourages the gating network to assign distinct expert combinations across threats, effectively constructing separate model pathways for each threat type.

Let $\mathcal{B}_p$ denote adversarial examples under threat $\mathcal{A}_p$ for $p \in \{1, 2, \infty\}$. For each threat p and token i, we compute the average local gating $\bar{\mathbf{u}}_i^p$ across $\mathcal{B}_p$. We then maximize pairwise Euclidean distances between threat-specific gating patterns, aggregating across all T token positions:

$$\mathcal{L}_{\text{div-local}} = -\frac{1}{T} \sum_{p \neq q} \sum_{i=1}^{T} \|\bar{\mathbf{u}}_i^p - \bar{\mathbf{u}}_i^q\|_2^2 . \tag{6}$$

Since global gating weights are K-dimensional (number of experts), we project them to a higher K' dimension via a linear layer for finer-grained threat discrimination. The average projected gating $\bar{\mathbf{v}}^p$ for threat p yields the following loss:

$$\mathcal{L}_{\text{div-global}} = -\sum_{p \neq q} \|\bar{\mathbf{v}}^p - \bar{\mathbf{v}}^q\|_2^2 . \tag{7}$$

This projection is used for computing diversification loss and does not affect routing in Eq. 4. In Table A5, we explore other options for measuring the distance between gating weights, out of which our design leads to the best performance.

We aggregate both objectives for each layer l across all L layers to encourage gating networks to learn distinct, threat-specific expert routing:

$$\mathcal{L}_{\text{div}} = \frac{1}{L} \sum_{l=1}^{L} \left(\mathcal{L}_{\text{div-local}}^{(l)} + \mathcal{L}_{\text{div-global}}^{(l)} \right). \tag{8}$$

The diverse model pathways learned by the experts provide broader coverage of the perturbation space, improving robustness against unseen threats compared to existing MAT methods that rely on a single model pathway. Furthermore, since threat labels are only used during training, the learned gating predicts the optimal expert combination at inference without requiring knowledge of the input threat type, allowing our method to generalize to unseen threats. Notably, this generalization extends beyond ℓ_p threats and improves robustness also against various non-ℓ_p threats (Table 2, Sec. 5.2).

4.5 Training Objective

Thanks to its modular design, RoME can be integrated with existing MAT methods. In this work, we apply it to MAX [59] and RANDOM [42]. The training objective combines the base loss (Eq. 1) with our diversification regularization:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\phi\left(\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_\infty\right)\right] + \lambda \mathcal{L}_{\text{div}}, \tag{9}$$

where ϕ denotes aggregation function ϕ_{MAX} or ϕ_{RANDOM}, and λ controls strength of $\mathcal{L}_{\text{div}}$. θ denotes parameters for backbone, experts, and gating networks.

While this section focuses on ℓ_p-MAT, modular design of our RoME makes it widely applicable to other adversarial training methods. In Sec. 5.2, we demonstrate this by applying RoME to non-ℓ_p perceptual adversarial training [13, 33] and show that RoME further improves their robustness against unseen threats.

5 Experiments

5.1 Experimental Setup

Datasets, models, and baselines. We evaluate on CIFAR-10 [30], ImageNet-100 [6], and ImageNet-1K [14] using standard ImageNet-1K pre-trained [14, 62] ViT-B [15], DeiT-B [58], and Swin-B [38]. For MAT, we compare with RANDOM [42], AVG [59], MAX [59], MSD [44], MORE [5], E-AT [9], and RAMP [28]. For evaluation against unseen threats, we follow OODRobustBench [36] and also compare with PAT [33] and VR [13], two representative non-ℓ_p adversarial training methods trained on AlexNet [31]-based and self-model perceptual threats. For all baselines, we use default hyperparameters from their original manuscripts.

Threat setup. Following existing protocols [9, 28, 59], we train and evaluate under PGD [43] and APGD [8] with perturbation budgets $(\epsilon_1, \epsilon_2, \epsilon_\infty)$ of $(12, 0.5, \frac{8}{255})$ for CIFAR-10 and $(255, 2.0, \frac{4}{255})$ for ImageNet-100, 1K. For PGD [42, 44, 59], we use steps (n_1, n_2, n_∞) of $(20, 20, 10)$ for CIFAR-10 and $(40, 20, 10)$ for ImageNet-100, 1K. For APGD [9,28], we use steps $n = 15$ for both CIFAR-10 and ImageNet-100, 1K. For robust fine-tuning, following E-AT [9] and RAMP [28], we use APGD with 10 steps on CIFAR-10 and 5 (ℓ_∞/ℓ_2) and 15 (ℓ_1) steps on ImageNet-1K. For evaluation, we use AutoAttack under ℓ_1, ℓ_2, and ℓ_∞ norms. We evaluate natural accuracy on clean images, per-threat robustness, their average, and worst-case union robustness across all threats [28].

Table 1: Comparison with state-of-the-art methods on CIFAR-10 and ImageNet-100 under AutoAttack. We measure natural accuracy on clean images, single-threat robustness under ℓ_1, ℓ_2, and ℓ_∞, their average, and worst-case union robustness. Best results are marked in **bold**.

Methods		*CIFAR-10*						*ImageNet-100*					
		Nat	ℓ_1	ℓ_2	ℓ_∞	Avg	Union	Nat	ℓ_1	ℓ_2	ℓ_∞	Avg	Union
PGD	RANDOM	86.7	52.3	68.8	40.3	53.8	38.6	84.4	43.8	67.6	45.2	52.2	39.4
	AVG	86.3	52.2	67.5	38.9	52.9	37.9	83.6	44.3	66.4	44.8	51.8	39.3
	MAX	85.1	46.6	66.8	40.1	51.2	38.5	79.9	42.2	63.2	47.5	51.0	40.5
	MSD	84.7	43.9	67.3	40.5	50.6	38.3	80.2	42.6	64.3	45.8	50.9	40.2
	MORE	82.6	44.3	42.4	33.7	40.1	31.1	75.6	34.5	41.2	42.0	39.2	32.8
	RoME+RANDOM (Ours)	**88.1**	**56.4**	**69.7**	42.1	**56.1**	41.1	**85.2**	**46.9**	**68.3**	47.8	**54.3**	42.9
	RoME+MAX (Ours)	85.4	48.2	68.2	**43.1**	53.2	**42.3**	80.6	45.6	66.0	**50.9**	54.2	**43.7**
APGD	RANDOM	86.5	50.1	69.3	37.5	52.3	37.2	82.0	38.4	66.0	46.2	50.2	37.1
	MAX	81.7	45.5	65.3	42.6	51.1	41.4	79.5	41.2	63.6	44.8	49.9	39.8
	E-AT	83.2	50.3	68.9	40.6	53.3	39.7	**83.4**	40.6	64.2	45.4	50.1	39.2
	RAMP	82.2	47.1	63.5	43.4	51.3	42.5	82.2	42.6	63.4	45.1	50.4	41.8
	RoME+RANDOM (Ours)	**87.4**	**54.7**	**71.6**	40.0	**55.4**	39.4	**83.4**	42.1	**67.8**	**48.6**	**52.8**	40.5
	RoME+MAX (Ours)	82.5	48.9	67.3	**44.2**	53.5	**43.7**	80.2	**44.2**	64.3	47.9	52.1	**42.8**

Implementation details. For fair comparison, we follow standard Transformer training recipe [15, 58] for all baselines and our method, using AdamW [39] with initial learning rates of `1e-3` for CIFAR-10 and `1e-4` for ImageNet-100, 1K, with warmup and linear decay scheduling (full details in Sec. A1). We train for 20 epochs on CIFAR-10 and 5 epochs on ImageNet-100, 1K. For robust fine-tuning, we fine-tune ℓ_∞-robust models for 3 epochs on CIFAR-10 and 1 epoch on ImageNet-1K. We use $K = 4$ experts of rank 16, apply them to QKV and O projection layers in all Transformer blocks, and set $\lambda = 0.1$, $s = 4$, $b = 2$, and $K' = 100$ for global gating projection. We also apply our method to CNNs (WideResNet [71]), where low-rank experts on convolutional layers are implemented using the official LoRA implementation [3], and local and global features are obtained per spatial location of the convolutional feature map and by global average pooling over all spatial locations, respectively.

5.2 Multi-Perturbation Adversarial Robustness

Robustness against multiple ℓ_p perturbations. In Table 1, we report natural accuracy and robustness against ℓ_p threats seen during training. Our approach effectively mitigates cross-threat trade-off, achieving the best average and union robustness across all settings. For example, our RoME+MAX improves union robustness of baseline MAX by 3.8%p on CIFAR-10 under PGD and 3.0%p on ImageNet-100 under APGD. We further validate on ImageNet-1K (Table A2), where our method improves union robustness by 1.7%p and 1.7%p over RANDOM and MAX under PGD and by 1.3%p and 1.6%p under APGD, demonstrating its scalability. It also consistently outperforms baselines

[3] `https://github.com/microsoft/LoRA/blob/main/loralib/layers.py`

Table 2: Comparison with prior methods on CIFAR-10 under unseen common corruptions and adversarial threats. Percep. ℓ_p is the average of Fog [33], Snow [33], Gabor [33], Elastic [33], and ℓ_∞-Jpeg [33] robustness. Best results are marked in **bold**.

	Methods	Com. Corr.	ℓ_0	Percep. ℓ_p	PPGD	LPA	Adv. Patch	StAdv	ReColorAdv	GMA
PGD	RANDOM	79.0	31.9	53.2	43.7	32.2	47.9	31.1	70.4	44.0
	AVG	78.4	26.7	54.5	42.9	31.0	48.6	29.2	68.8	43.7
	MAX	76.8	28.3	51.3	43.2	33.1	43.9	34.4	70.0	45.4
	MSD	76.5	28.0	50.4	42.4	33.3	40.8	34.9	70.8	45.6
	MORE	74.9	23.8	50.7	42.7	31.5	45.2	32.1	70.0	43.6
	RoME+RANDOM (Ours)	**80.2**	38.3	**57.8**	51.2	35.4	**48.8**	36.8	73.5	45.6
	RoME+MAX (Ours)	76.8	**39.4**	56.2	**52.2**	**40.7**	47.4	**44.0**	**75.4**	**47.5**
APGD	RANDOM	78.5	31.6	54.5	49.5	45.5	49.1	47.0	77.8	40.0
	MAX	74.2	33.4	51.6	50.5	49.2	47.2	54.6	76.6	44.9
	E-AT	74.3	34.1	49.7	49.7	47.9	48.0	48.5	75.5	42.8
	RAMP	74.9	33.1	51.6	51.0	50.0	49.4	49.9	76.5	45.1
	RoME+RANDOM (Ours)	**79.6**	**38.8**	**58.9**	50.9	47.0	**50.4**	49.2	**78.2**	43.3
	RoME+MAX (Ours)	74.5	37.8	53.0	**51.5**	**51.6**	49.2	**56.0**	78.0	**45.6**
Non-ℓ_p	PAT	77.6	16.0	47.1	55.5	51.5	41.2	59.7	76.4	49.0
	PAT+VR	77.7	9.9	47.9	56.6	51.9	39.3	58.4	77.0	50.1
	RoME+PAT (Ours)	78.1	**19.8**	48.6	57.0	54.9	45.2	63.3	77.5	49.8
	RoME+PAT+VR (Ours)	**78.3**	17.9	**49.1**	**59.3**	**57.8**	**45.7**	**65.2**	**80.1**	**52.6**

on DeiT-B [58] and Swin-B [38] (Table A1). Our approach also achieves the highest natural accuracy. This is because prior MAT methods force all inputs into a single representation space, where the accuracy-robustness trade-off inherent in adversarial training [60, 72] degrades performance on clean images. In contrast, our RoME learns a unique model pathway for natural images differently from adversarial examples as shown in Fig. A2 (Sec. A3.2).

Robustness against unseen attacks. Following previous benchmarks [9, 28], we evaluate on common corruptions [22], ℓ_0 attack [46], ℓ_p perceptual attacks [33], and AutoAttack [8]. We also evaluate on non-ℓ_p threats, including perceptual PPGD [33] and LPA [33], adversarial patch [2], spatial StAdv [65], and semantic ReColorAdv [32] attacks. We also design and evaluate against gating misrouting attack (GMA), a strong white-box adaptive attack with full access to the gating framework. GMA optimizes $\mathcal{L}_{\text{GMA}} = \mathcal{L}_{\text{CE}}(f_\theta(x+\delta), y) - \frac{1}{L}\sum_{l=1}^{L}\frac{1}{T}\sum_{i=1}^{T}\mathcal{L}_{\text{CE}}(\mathbf{g}_i^{(l)}(x+\delta), k_i^*)$, where the first term maximizes the classification loss and the second forces misrouting of each token's gating weight $\mathbf{g}_i^{(l)}$ (Eq. 5) to the least likely expert $k_i^* = \arg\min_k g_{i,k}$. For this attack, we use ℓ_∞-PGD with $\epsilon = \frac{8}{255}$ and 20 steps. Full attack configurations (budget and steps) are provided in Sec. A1.

Table 2 shows that our approach outperforms existing MAT methods across all unseen threats. Notably, while MORE [5] also leverages MoE, it shows lower robustness, highlighting that our approach of learning distinct model pathways improves generalization to unseen threats. Our method also outperforms existing methods against non-ℓ_p threats (*e.g.*, PPGD and LPA), demonstrating that the diverse representations captured across multiple model pathways generalize

Table 3: Comparison with existing methods on robust fine-tuning scenario across different model architectures pre-trained on ℓ_∞-adversarial training with different datasets. Best results are marked in **bold**.

	ViT-B (CIFAR-10)		WRN-28-10 (CIFAR-10)		WRN-94-16 (CIFAR-10)		XCiT-S (ImageNet-1K)	
Method	Nat	Union	Nat	Union	Nat	Union	Nat	Union
E-AT	83.6	41.7	89.4	50.3	92.8	44.7	65.0	25.9
+RoME	**84.7**	**43.5**	**91.3**	**51.8**	**93.3**	**48.4**	**66.1**	**27.6**
MAX	82.9	43.1	88.7	51.4	91.2	51.4	64.1	27.2
+RoME	**83.6**	**44.4**	**90.5**	**53.7**	**95.2**	**57.5**	**65.1**	**30.5**
RAMP	**83.1**	43.7	89.2	52.3	90.7	55.8	62.8	29.4
+RoME	82.7	**46.5**	**91.3**	**54.2**	**93.6**	**58.7**	**64.2**	**32.1**

beyond the ℓ_p threats seen during training. It maintains highest robustness even against adaptive attacks, verifying the robustness of our gating framework.

To demonstrate the modularity of our RoME (Sec. 4.5), we also compare with PAT [33] and VR [13], two representative non-ℓ_p adversarial training methods designed for unseen threat robustness. Different from ℓ_p-MAT, we follow the protocol of OODRobustBench [36] and train model on AlexNet [31]-based and self-model perceptual threats [33]. Applying our method on these methods consistently improves unseen threat robustness with gains of up to 6.8%p against StAdv, demonstrating that RoME is broadly applicable beyond ℓ_p-MAT.

Robust fine-tuning. In addition to training the model from scratch, we evaluate our method on robust fine-tuning scenario following E-AT [9]. For CIFAR-10, we use ViT-B-16 pre-trained with adversarial training on ℓ_∞ threats [43] along with WRN-28-10 [71] and WRN-94-16 [71] from RobustBench [7]. For ImageNet-1K, we use XCiT-S [1] also from RobustBench. As shown in Table 3, applying RoME to fine-tuning on ℓ_∞-robust models consistently improves union AutoAttack robustness across all models and baseline methods. For example, applying RoME to RAMP reaches 58.7% union robustness on WRN-94-16 (+14.0%p vs. E-AT, +2.9%p vs. RAMP), with similar trends on WRN-28-10 and XCiT-S. Beyond robustness, RoME also improves natural accuracy in most settings (*e.g.*, +4.0%p on WRN-94-16 with MAX). These consistent gains across ViT, CNN (WRN), and XCiT backbones show that RoME is architecture-agnostic.

5.3 Ablation Studies

In Table 4, we provide ablations on individual components of our RoME.

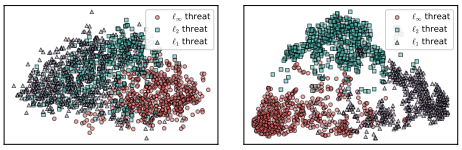
Low-rank experts. We verify that FFN experts leads to degraded union robustness, confirming that it is vital to capture threat-common features with a shared backbone. Our low-rank experts also outperforms adapter [23]-based experts, verifying the effectiveness of our design choices. In Table A4, we analyze the effect of applying RoME to different layers and find that applying experts to QKV and O projection layers achieves the best efficiency-robustness trade-off.

Table 4: Ablation studies on individual components of RoME on CIFAR-10. ‡ indicates using 3 experts for a fair comparison with gating classification, which requires one expert per threat; others use 4 experts. Best results are marked in **bold**.

Methods	Clean	ℓ_1	ℓ_2	ℓ_∞	Avg.	Union
Low-rank experts (Sec. 4.2)						
Expert architectures:						
FFN-based experts	76.5	46.4	58.6	31.5	45.5	30.9
Adapter-based experts	88.0	55.2	**70.6**	40.2	55.3	38.6
Ours (Low-rank experts)	**88.1**	**56.4**	69.7	**42.1**	**56.1**	**41.1**
Dual-scale gating (Sec. 4.3)						
No learned gating:						
No gating (uniform weights)	87.1	53.7	69.1	38.3	53.7	38.1
Single-scale gating:						
Local (token-level) only	87.5	53.1	69.6	39.9	54.2	39.4
Global (image-level) only	87.0	55.0	69.3	40.2	54.8	39.8
Dual-scale gating variants:						
Inverted layer weights	87.6	55.2	69.4	39.7	54.8	39.3
Learned layer weights	87.6	53.7	69.5	40.3	54.5	39.6
CLS token for global	87.8	54.5	69.2	39.0	54.2	38.7
Ours (avg. + adaptive)	**88.1**	**56.4**	**69.7**	**42.1**	**56.1**	**41.1**
Threat-guided gating diversification loss (Sec. 4.4)						
Conventional MoE regularization:						
No regularization	86.7	52.9	69.3	38.7	53.6	38.1
Load balancing [56]	86.9	55.0	69.5	38.8	54.4	38.4
Importance [56]	87.0	53.5	69.3	39.0	53.9	38.6
Threat-aware regularization:						
Gating classification‡	87.5	42.9	69.8	34.4	49.0	31.4
Global gating projection:						
w/o projection layer	87.7	56.7	**69.9**	41.6	56.1	40.6
Ours (diversification)‡	87.9	**57.4**	69.5	41.9	**56.3**	41.0
Ours (diversification)	**88.1**	56.4	69.7	**42.1**	56.1	**41.1**

Dual-scale gating. Removing learned gating degrades robustness, showing that threat-specific expert selection is critical. Single-scale gatings underperform our approach, verifying that different granularities are required for threat discrimination (Fig. 5). While inverting or learning the layer-adaptive weights (Eq. 5) improves robustness thanks to dual-scale gating, both remain suboptimal, verifying that encoding the Transformer layer hierarchy [20,37] as a fixed prior is more effective than making it learnable. Replacing averaged global feature with CLS token degrades performance, verifying the effectiveness of our design choices.

Gating diversification. Applying no regularization, load balancing [56], or importance loss [56] leads to suboptimal robustness due to threat-agnostic routing. Gating classification, which trains the gating network as a threat classifier with cross-entropy loss, remains suboptimal as it maps each threat to a single fixed expert unlike our flexible diversification loss. Removing the projection layer in



(a) Backbone **(b)** Backbone + experts

Fig. 6: t-SNE visualization of (a) backbone features and (b) backbone + expert outputs from the final ViT-B layer.

Table 5: Analysis on learned experts and gating during inference with RANDOM and PGD training on CIFAR-10. Best results are marked in **bold**.

Methods	Clean	ℓ_1	ℓ_2	ℓ_∞	Avg.	Union
Random gating	88.0	55.1	69.5	39.3	54.6	38.4
Inverse gating	87.8	55.4	69.0	39.6	54.7	38.9
w/o experts	**88.1**	54.9	69.4	39.9	54.7	39.2
Ours	**88.1**	**56.4**	**69.7**	**42.1**	**56.1**	**41.1**

global diversification reduces union robustness to 40.6%, and alternative distance choices in Table A5 all underperform our method. Fig. A3 further shows that our method is not overly sensitive to hyperparameters including expert rank, number of experts, loss weight λ, and s, b for layer-adaptive weighting.

5.4 Analysis on Experts and Threat-Aware Gating

Learned features on backbone vs. experts. To verify whether low-rank experts learn threat-specific features, we analyze t-SNE visualizations of backbone features and expert outputs added to the backbone in Fig. 6. Backbone captures similar features regardless of threat type as shown in Fig. 6a, showing that it has captured features common among threats. In contrast, features passed through our low-rank experts capture decoupled representations for each threat as shown in Fig. 6b, showing that our experts have learned threat-specific features.

Learned routing for ℓ_p threats. To understand how threats affect expert routing, we apply different threats to various spatial regions and visualize the resulting expert activations for each image patch. As shown in Fig. 7, standard MoE shows similar expert activations for all image patches (Expert 1 in row 1, left), verifying that experts are not specialized for a specific threat type. In contrast, our RoME learns threat-specific routing; for example, Expert 1 (row 2) is highly activated for ℓ_∞ threats while barely activated for ℓ_1 threats, demonstrating that each expert specializes in a distinct threat type.

Analysis on experts and gating. We analyze learned experts and gating through inference-time variants (Table 5). Random weights, inverted gating $1-\mathbf{g}$, or removing experts degrades robustness, verifying that gating learns threat-specific combinations and that experts learn specialized representations.

Computational analysis. In Table 6, we compare computational costs with baseline RANDOM [42]. MORE [5] and conventional MoE [56] with FFN experts incur 4× and 3× more parameters, with 8.25× and 2.75× more train time compared to RANDOM. In contrast, our method achieves SoTA union robustness (Table 1) with only 1.04× more parameters and 1.17× more train time. Our method also shows superior optimization efficiency, consistently outperforming robustness of RANDOM given the same wall-clock training time (Fig. 8). It also reaches RANDOM's peak robustness in 13.2k sec of training versus 23.9k sec for

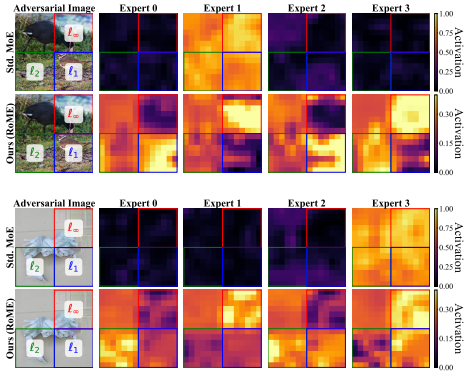


Fig. 7: Expert activations for various ℓ_p threats applied to different regions of image. While standard MoE (rows 1, 3) routes threats to same experts, RoME (rows 2, 4) utilizes different experts for each threat.

Table 6: Computational cost comparison. Best results are marked in **bold**.

Methods	# Params (M)	FLOPs (G)	Train time (sec/iter)	Latency (ms/img)
RANDOM [42]	85.15	5.61	3.05	0.53
MORE [5]	340.60	22.43	25.16	1.70
MoE (FFN) [56]	255.20	9.29	8.38	1.36
Ours	**88.85**	**5.84**	**3.56**	**0.60**

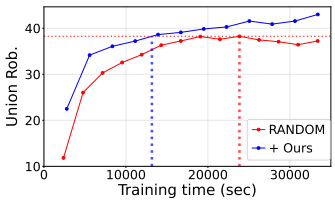


Fig. 8: Training time vs. robustness. Ours consistently outperforms baseline given the same wall-clock training time.

RANDOM. Baselines with additional expert and gating layers but without our training components in Table A3 show barely improved robustness, showing that simply increasing model capacity does not address the cross-threat trade-off.

6 Conclusion

We introduce RoME, a novel framework that addresses cross-threat robustness trade-off in multi-perturbation adversarial training through mixture of low-rank experts. We identify the threat-agnostic routing problem where conventional mixture of experts faces difficulty distinguishing between threats and learns similar routing. To address this, we propose threat-distinguishing dual-scale gating and threat-guided gating diversification that enable distinct expert combinations for each threat. Extensive experiments demonstrate that RoME achieves state-of-the-art union robustness across diverse benchmarks, while also improving robustness against a diverse set of unseen threats. Ablation studies and analysis demonstrate the ability of our method to effectively distinguish between threats and learn diverse threat-specific pathways, leading to improved robustness.

Acknowledgement. Sung-eui Yoon is the corresponding author. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (RS-2023-00237965, Recognition, Action and Interaction Algorithms for Open-world Robot Service) and the IITP(Institute of Information & Communications Technology Planning & Evaluation)-ITRC(Information Technology Research Center) grant funded by the Korea government(Ministry of Science and ICT)(IITP-2026-RS-2020-II201460).

References

1. Ali, A., Touvron, H., Caron, M., Bojanowski, P., Douze, M., Joulin, A., Laptev, I., Neverova, N., Synnaeve, G., Verbeek, J., et al.: Xcit: Cross-covariance image transformers. In: NeurIPS (2021)
2. Brown, T.B., Mané, D., Roy, A., Abadi, M., Gilmer, J.: Adversarial patch. In: NeurIPS (2017)
3. Chen, T., Chen, X., Du, X., Rashwan, A., Yang, F., Chen, H., Wang, Z., Li, Y.: Adamv-moe: Adaptive multi-task vision mixture-of-experts. In: ICCV (2023)
4. Chen, Z., Shen, Y., Ding, M., Chen, Z., Zhao, H., Learned-Miller, E.G., Gan, C.: Mod-squad: Designing mixtures of experts as modular multi-task learners. In: CVPR (2023)
5. Cheng, H., Xu, K., Wang, C., Kailkhura, B., Lin, X., Goldhahn, R.: Mixture of robust experts (more): A robust denoising method towards multiple perturbations. arXiv preprint arXiv:2104.10586 (2021)
6. Chun-Hsiao Yeh, Y.C.: IN100pytorch: Pytorch implementation: Training resnets on imagenet-100. `https://github.com/danielchyeh/ImageNet-100-Pytorch` (2022), accessed: 2026-03-04
7. Croce, F., Andriushchenko, M., Sehwag, V., Debenedetti, E., Flammarion, N., Chiang, M., Mittal, P., Hein, M.: Robustbench: a standardized adversarial robustness benchmark. In: NeurIPS Datasets and Benchmarks Track (2021)
8. Croce, F., Hein, M.: Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: ICML (2020)
9. Croce, F., Hein, M.: Adversarial robustness against multiple and single l_p-threat models via quick fine-tuning of robust classifiers. In: ICML (2022)
10. Croce, F., Rebuffi, S.A., Shelhamer, E., Gowal, S.: Seasoning model soups for robustness to adversarial and natural distribution shifts. In: CVPR (2023)
11. Dai, D., Deng, C., Zhao, C., Xu, R., Gao, H., Chen, D., Li, J., Zeng, W., Yu, X., Wu, Y., et al.: Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. arXiv preprint arXiv:2401.06066 (2024)
12. Dai, S., Cianfarani, C., Bhagoji, A., Sehwag, V., Mittal, P.: Adapting to evolving adversaries with regularized continual robust training. In: ICML (2025)
13. Dai, S., Mahloujifar, S., Mittal, P.: Formulating robustness against unforeseen attacks. In: NeurIPS (2022)
14. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
15. Dosovitskiy, A.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
16. Dou, S., Zhou, E., Liu, Y., Gao, S., Shen, W., Xiong, L., Zhou, Y., Wang, X., Xi, Z., Fan, X., et al.: Loramoe: Alleviating world knowledge forgetting in large language models via moe-style plugin. In: ACL (2024)
17. Fan, Z., Sarkar, R., Jiang, Z., Chen, T., Zou, K., Cheng, Y., Hao, C., Wang, Z., et al.: M^3vit: Mixture-of-experts vision transformer for efficient multi-task learning with model-accelerator co-design. In: NeurIPS (2022)
18. Fedus, W., Zoph, B., Shazeer, N.: Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. Journal of Machine Learning Research (2022)
19. Gao, C., Chen, K., Rao, J., Liu, R., Sun, B., Zhang, Y., Peng, D., Guo, X., Subrahmanian, V.: Mola: Moe lora with layer-wise expert allocation. In: NAACL (2025)

20. Geva, M., Schuster, R., Berant, J., Levy, O.: Transformer feed-forward layers are key-value memories. In: EMNLP (2021)
21. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: ICLR (2015)
22. Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations. In: ICLR (2019)
23. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., Gelly, S.: Parameter-efficient transfer learning for nlp. In: ICML (2019)
24. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. In: ICLR (2022)
25. Huang, C., Liu, Q., Lin, B.Y., Pang, T., Du, C., Lin, M.: Lorahub: Efficient cross-task generalization via dynamic lora composition. In: COLM (2023)
26. Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., Madry, A.: Adversarial examples are not bugs, they are features. In: NeurIPS (2019)
27. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E.: Adaptive mixtures of local experts. Neural computation (1991)
28. Jiang, E., Singh, G.: Ramp: Boosting adversarial robustness against multiple l_p perturbations for universal robustness. In: NeurIPS (2024)
29. Jung, M.J., Kim, J.: Pmoe: Progressive mixture of experts with asymmetric transformer for continual learning. arXiv preprint arXiv:2407.21571 (2024)
30. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
31. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NeurIPS (2012)
32. Laidlaw, C., Feizi, S.: Functional adversarial attacks. In: NeurIPS (2019)
33. Laidlaw, C., Singla, S., Feizi, S.: Perceptual adversarial robustness: Defense against unseen threat models. In: ICLR (2021)
34. Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., Krikun, M., Shazeer, N., Chen, Z.: Gshard: Scaling giant models with conditional computation and automatic sharding. In: ICLR (2021)
35. Li, H., Lin, S., Duan, L., Liang, Y., Shroff, N.B.: Theory on mixture-of-experts in continual learning. In: ICLR (2025)
36. Li, L., Wang, Y., Sitawarin, C., Spratling, M.: Oodrobustbench: a benchmark and large-scale analysis of adversarial robustness under distribution shift. In: ICML (2024)
37. Liao, M., Chen, W., Shen, J., Guo, S., Wan, H.: Hmora: Making llms more effective with hierarchical mixture of lora experts. In: ICLR (2025)
38. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: ICCV (2021)
39. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: ICLR (2019)
40. Ma, J., Zhao, Z., Yi, X., Chen, J., Hong, L., Chi, E.H.: Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In: KDD (2018)
41. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. In: JMLR (2008)
42. Madaan, D., Shin, J., Hwang, S.J.: Learning to generate noise for multi-attack robustness. In: ICML (2021)
43. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: ICLR (2018)
44. Maini, P., Wong, E., Kolter, Z.: Adversarial robustness against the union of multiple perturbation models. In: ICML (2020)

45. Meymani, M., Razavi-Far, R.: Defending against adversarial attacks using mixture of experts. arXiv preprint arXiv:2512.20821 (2025)
46. Modas, A., Moosavi-Dezfooli, S.M., Frossard, P.: Sparsefool: a few pixels make a big difference. In: CVPR (2019)
47. Mu, S., Lin, S.: A comprehensive survey of mixture-of-experts: Algorithms, theory, and applications. arXiv preprint arXiv:2503.07137 (2025)
48. Pavlitska, S., Eisen, E., Zöllner, J.M.: Towards adversarial robustness of model-level mixture-of-experts architectures for semantic segmentation. In: 2024 International Conference on Machine Learning and Applications (ICMLA) (2024)
49. Pavlitska, S., Fan, H., Ditschuneit, K., Zöllner, J.M.: Robust experts: the effect of adversarial training on cnns with sparse mixture-of-experts layers. In: ICCV (2025)
50. Puigcerver, J., Jenatton, R., Riquelme, C., Awasthi, P., Bhojanapalli, S.: On the adversarial robustness of mixture of experts. In: NeurIPS (2022)
51. Puigcerver, J., Riquelme, C., Mustafa, B., Houlsby, N.: From sparse to soft mixtures of experts. In: ICLR (2024)
52. Riquelme, C., Puigcerver, J., Mustafa, B., Neumann, M., Jenatton, R., Susano Pinto, A., Keysers, D., Houlsby, N.: Scaling vision with sparse mixture of experts. In: NeurIPS (2021)
53. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics (1987)
54. Rusu, A.A., Rabinowitz, N.C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., Hadsell, R.: Progressive neural networks. arXiv preprint arXiv:1606.04671 (2016)
55. Shafahi, A., Najibi, M., Ghiasi, M.A., Xu, Z., Dickerson, J., Studer, C., Davis, L.S., Taylor, G., Goldstein, T.: Adversarial training for free! In: NeurIPS (2019)
56. 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. In: ICLR (2017)
57. Singh, N.D., Croce, F., Hein, M.: Revisiting adversarial training for imagenet: Architectures, training and generalization across threat models. In: NeurIPS (2023)
58. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: ICML (2021)
59. Tramer, F., Boneh, D.: Adversarial training and robustness for multiple perturbations. In: NeurIPS (2019)
60. Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., Gu, Q.: Improving adversarial robustness requires revisiting misclassified examples. In: ICLR (2020)
61. Wang, Z., Li, X., Zhu, H., Xie, C.: Revisiting adversarial training at scale. In: CVPR (2024)
62. Wightman, R.: Pytorch image models. `https://github.com/rwightman/pytorch-image-models` (2019). `https://doi.org/10.5281/zenodo.4414861`
63. Wong, E., Rice, L., Kolter, J.Z.: Fast is better than free: Revisiting adversarial training. In: ICLR (2020)
64. Wu, X., Huang, S., Wei, F.: Mixture of lora experts. In: ICLR (2024)
65. Xiao, C., Zhu, J.Y., Li, B., He, W., Liu, M., Song, D.: Spatially transformed adversarial examples. In: ICLR (2018)
66. Xie, C., Yuille, A.: Intriguing properties of adversarial training at scale. In: ICLR (2020)
67. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al.: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025)

68. Yang, Y., Qin, Y., Wu, T., Xu, Z., Li, G., Guo, P., Shao, H., Shi, Y., Li, K., Sun, X., et al.: Leveraging open knowledge for advancing task expertise in large language models. arXiv preprint arXiv:2408.15915 (2024)
69. Yang, Y., Jiang, P.T., Hou, Q., Zhang, H., Chen, J., Li, B.: Multi-task dense prediction via mixture of low-rank experts. In: CVPR (2024)
70. Yu, J., Zhuge, Y., Zhang, L., Hu, P., Wang, D., Lu, H., He, Y.: Boosting continual learning of vision-language models via mixture-of-experts adapters. In: CVPR (2024)
71. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: BMVC (2016)
72. Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., Jordan, M.: Theoretically principled trade-off between robustness and accuracy. In: ICML (2019)
73. Zhang, X., Xu, K., Hu, Z., Wang, R.: Optimizing robustness and accuracy in mixture of experts: A dual-model approach. In: ICML (2025)
74. 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: ICCV (2023)
75. Zhao, X., Chen, X., Cheng, Y., Chen, T.: Sparse moe with language guided routing for multilingual machine translation. In: ICLR (2024)
76. Zhou, H., Wang, Z., Huang, S., Huang, X., Han, X., Feng, J., Deng, C., Luo, W., Chen, J.: Moe-lpr: Multilingual extension of large language models through mixture-of-experts with language priors routing. In: AAAI (2025)
77. Zhou, Y., Lei, T., Liu, H., Du, N., Huang, Y., Zhao, V., Dai, A.M., Le, Q.V., Laudon, J., et al.: Mixture-of-experts with expert choice routing. In: NeurIPS (2022)
78. Zhou, Z., Firestone, C.: Humans can decipher adversarial images. Nature communications (2019)