CoMoE: Contrastive Mixture-of-Experts are Efficient Representation Learners

Anonymous authors
Paper under double-blind review

Abstract

While Contrastive Learning (CL) achieves great success in many downstream tasks, its good performance heavily relies on a large model capacity. As previous methods focus on scaling dense models, training and inference costs increase rapidly with model sizes, leading to large resource consumption. In this paper, we explore CL with an efficient scaling method, Mixture of Experts (MoE), to obtain a large but sparse model. We start by plugging in the state-of-the-art CL method to MoE. However, this naive combination fails to visibly improve performance despite a much larger capacity. A closer look reveals that the naive MoE+CL model has a strong tendency to route two augmented views of the same image token to different subsets of experts: such "cross-view instability" breaks the weight-sharing nature in CL and misleads the invariant feature learning. To address this issue, we introduce a new regularization mechanism, by enforcing expert-routing similarity between different views of the same image (or its overlapped patch tokens), while promoting expert-routing diversity of patches from different images. The resultant method, called CoMoE, improves by 1.7 points in terms of 1% semi-supervised learning accuracy on ImageNet, compared to the naive combination baseline. It further surpasses the state-of-the-art CL methods on ImageNet pre-training of Vision Transformer (ViT) by 2.8 points, at the same computational cost. Our findings validate CoMoE as an effective and efficient image representation learner. Code is included in the supplemental materials.

1 Introduction

Unsupervised contrastive Learning (CL) has been popularly explored as it demonstrate strong performance on many downstream tasks, which could even beat its supervised counterpart (Chen et al., 2020c; Grill et al., 2020; Caron et al., 2020; Chen et al., 2021b; Caron et al., 2021). However, the performance of CL heavily relays on the large capacity of the employed model. For instance, in semi-supervised learning with few labels, one important application of self-supervised learning (Tian et al., 2020b), SimCLR-v2 (Chen et al., 2020b) demonstrates that scaling model parameters from 24M to 795M brings a performance improvement by 17%. However, scaling dense models significantly increases the training and inference cost. For instance, The 795M model would increase the training time by 41 times, and training to full performance (1000 epochs) on ImageNet-1K (Deng et al., 2009) requires 7000 GPU (V100) days.

In this paper, we study employing an efficient scaling method, a sparse Mixture of Experts

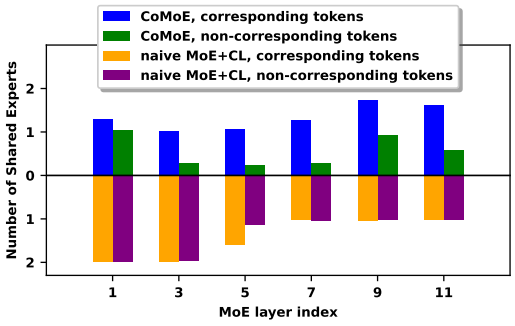


Figure 1: Routing comparison between the proposed CoMoE (upper two bars) and naive MoE+CL (the lower two bars). The X-axis denotes the layer index of the MoE layer. Y-axis is the Number of Shared Experts (NoSE) between a pair of randomly sampled tokens. The corresponding and non-corresponding tokens denote tokens from the same image and different images, respectively. The naive MoE+CL yields a small difference for the NoSE between corresponding and non-corresponding tokens, which can result in collapsing to the same experts or (partially) non-shared weight contrastive learning, thus a performance drop. The proposed CoMoE addresses this by enlarging the difference. More details in Section 4.2.

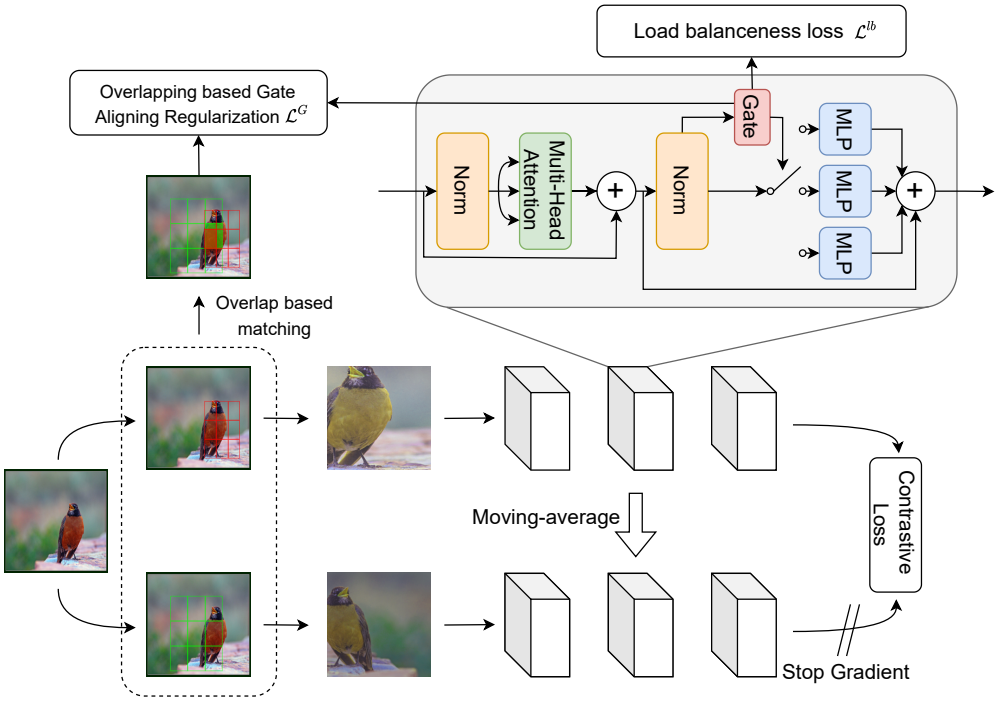


Figure 2: Pipeline of the proposed CoMoE. It replaces every other block of ViT to sparse MoE layer. Overlapping based gate aligning regularization is applied for training the proposed network.

(MoE) (Shazeer et al., 2017), for CL, without sacrificing training and inference efficiency. In contrast to dense models that process each sample with all parameters, MoE leverages a dynamic sparse model: each sample is routed to a small subset of experts. In this way, a large candidate pools of experts can be built while only activating a small part for each sample, making it possible to leverage large model capacity while maintaining small computational costs for training and inference. MoE has been applied successfully in NLP applications (Lepikhin et al., 2020; Fedus et al., 2021) and was recently introduced to vision tasks but only for supervised settings (Riquelme et al., 2021).

We start with directly applying CL on vision MoE models (e.g. Riquelme et al. (2021)). However, we find this naive combination only yields marginal performance improvement compared to its dense counterpart despite a much larger capacity. Looking closer, we observe that different augmented views of the same image tokens are mostly routed to different subsets of experts (as illustrated in Figure 1), This essentially breaks the conventional design of contrasting **shared weight branches** (Chen et al., 2020a; He et al., 2020; Grill et al., 2020) and turns to contrasting **independent branches**, which we show hurts performance with further empirical evaluations.

To enforce consistency in expert selections for augmented image views, a naive way is to always assign to them the same set of experts. However, this leaks the learning target of CL: the instance identity, causing the model to overfit on such trivial nuisance without learning meaningful image representations (Chen et al., 2021a). Instead, as shown in Figure 2, we propose a simple yet effective regularization mechanism to enforce the consistency of expert selection based on visual overlapping. Specifically, first we pair all image tokens based on the overlapping between patches. Then we pull the selection of experts of paired tokens to be similar while differentiating that for tokens from different images through the proposed Overlapping-based Gate Aligning Regularization (OGAR). The resulting method, termed CoMoE, significantly improves the consistency of the experts selection for different augments of the same image (as shown in Figure 1) and the 1% semi-supervised performance by 1.7 points compared to the naive plugin, which is also 2.8 points higher than competing state-of-the-art CL methods on ViT.

Our contributions are summarized as follows:

- We propose CoMoE, which efficiently scales Contrastive Learning (CL) with the sparse Mixture of Experts, pushing the limit of CL towards large model capacity while maintaining similar computation cost.
- We identify the problem of naively combining MoE and CL, which essentially routes semantically similar images to different sets of expert thus hurting performance, and address it through a novel overlapping-based regularization framework for all paired image tokens.
- Extensive experiments verifies the effectiveness of the proposed regularization term. Compared to competitive state-of-the-art CL methods on ViT, the proposed CoMoE achieves an improvement of 2.8 points at the same computational cost.

2 Related works

2.1 Self-supervised training

Inspired by the observation that conducting instance recognition could yield a good representation that naturally clusters the same class images (Alexey et al., 2016; Wu et al., 2018), various works devote to designing self-supervised learning through pulling the representations of the same images together while pushing those of different images apart (Chen et al., 2020c;a; He et al., 2020; Tian et al., 2020a), also known as contrastive learning. Some works also recognize that negative samples are not necessary (Grill et al., 2020; Misra & Maaten, 2020; Chen & He, 2021; Zbontar et al., 2021). A trend was observed and verified by (Chen et al., 2020a;b) that contrastive learning yields better performance with a longer training schedule and a large backbone model. However, training large models with CL for a long schedule imposes significantly high training costs. In this work, to scale CL we investigate an efficient scaling option based on Mixture-of-Experts(MoE). While recent work (Meng et al., 2022) also starts to explore sparsifying the contrastive learning with dynamic pruning strategies, MoE has its unique strength on memory efficiency and combining it with contrastive learning is still not explored.

Other works on self-supervised learning focus on the handcrafted pretext tasks (Trinh et al., 2019) like rotation prediction (Gidaris et al., 2018), jigsaw (Noroozi & Favaro, 2016; Carlucci et al., 2019) and colorization (Gidaris et al., 2018). Recent advances in transformer highlight the possibility of a new class of self-supervised learning methods through masked image modeling (Bao et al., 2021; He et al., 2021b; Xie et al., 2021). These conceptually different directions can also be combined with contrastive learning to further boosting the performance (Dangovski et al., 2021; Zhou et al., 2021). In this work, we focus on studying contrastive learning while leaving other directions as potential future work.

2.2 Sparse Mixture of Experts

The traditional Mixture of Experts Network is composed of multiple sub-models and conduct input conditional computation (Jacobs et al., 1991; Jordan & Jacobs, 1994; Chen et al., 1999; Yuksel et al., 2012; Roller et al., 2021). While contrastive learning can also be improved with the traditional MoE (Tsai et al., 2020), it suffers from intensive computation since the model are dense and all experts are activated. Recent work (Shazeer et al., 2017) proposes the Sparse Mixture of Experts Layer and demonstrates better results on language modeling with lower computational cost. Following works devise methods to further address the communication cost (Fedus et al., 2021; Lewis et al., 2021) and stability (Zoph et al., 2022) issues. GLaM (Du et al., 2021) studies the MoE for language self-supervised task and achieve significant downstream few-shot performance.

MoE is recently applied for computer vision tasks (Riquelme et al., 2021; Gross et al., 2017; Xue et al., 2021; Wang et al., 2020; Tsai et al., 2018; Ahmed et al., 2016; Yang et al., 2019; Pavlitskaya et al., 2020). However, these works mostly focus only on supervised or weakly supervised learning. In this work, we aim to investigate applying MoE on self-supervised learning settings and related vision tasks.

3 METHOD

3.1 PRELIMINARIES

Contrastive learning Contrative learning is a self-supervised method via maximizing instance discriminativeness. For example, it enforces the similarity of positive pairs while enlarging the distance of negative pairs (Wu et al., 2018):

$$\mathcal{M}(v_i, v_i^+, V^-, \tau) = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp\left(v_i \cdot v_i^+/\tau\right)}{\exp\left(v_i \cdot v_i^+/\tau\right) + \sum_{v_i^- \in V^-}\exp\left(v_i \cdot v_i^-/\tau\right)} \tag{1}$$

where v_i^+ is considered a positive sample of sample v_i while the set V^- consists of negative samples. $\exp\left(v_i \cdot v_i^+/\tau\right)$ measures the similarity of positive pair (v_i, v_i^+) while $\exp\left(v_i \cdot v_i^-/\tau\right)$ measure the similarity of negative pair (v_i, v_i^-). τ is the temperature controlling the magnitude of all terms.

MoCo-v3 (Chen et al., 2021b) is one of the state-of-the-art self-supervised methods devised for ViT (Dosovitskiy et al., 2020). It encodes two crops C_1 and C_2 for each image under random data augmentation. The images are then encoded with network and its Exponential Moving Average (EMA). MoCo-v3 also introduce random token projection to stabilize the learning process. The loss of MoCo-v3 is defined as

$$L_{\mathrm{CL}} = \mathcal{M}(f_1, f_2, \{f\}^-, \tau) \tag{2}$$

where the the features (f_1, f_2) of two random views from the same image (C_1, C_2) are employed as positive samples while negative set $\{f\}^-$ is composed by the features of views from other images.

Sparse Mixture of Experts MoE reduces the computational cost via activating a small subset of computational graph for each sample. The basic building block of MoE is the sparse MoE layers, which consists of n_e FFN expert networks (E_1,E_2,$\cdots$,E_{n_e}). Formally, a MoE layer is defined as

$$y = \sum_{i=1}^{n_e} G(x)_i E_i(x) \tag{3}$$

where x and y are the input and output, respectively. G is the gating function that outputs a vector containing scores for each expert network $E_i(x)$, typically instantiated with a Softmax. By picking the top-k scored experts ($k << n_e$), the model only activates a small subset of expert networks for each sample. For G, we employ the noisy top-k gating design introduced in (Riquelme et al., 2021) as

$$G(x) = \mathrm{TopK}(\mathrm{Softmax}(Wx + \epsilon), k) \tag{4}$$

where W is a learnable weight and ϵ denotes Gaussian noise sampled from $\mathcal{N}\left(0, \frac{1}{n_e^2}\right)$. Wx controls the clean score of the gating function while noise in ϵ benefits the load balancing between experts. The sum of the score are then normalized with $\mathrm{Softmax}$ function and sparsified with TopK defined as

$$\mathrm{TopK}(v, k)_i = \begin{cases} v_i & \text{if } v_i \text{ is in the top } k \text{ elements of } v \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

In this work, we focus on studying applying MoE for the ViT (Dosovitskiy et al., 2020) backbone. We follow the strategy of the (Riquelme et al., 2021) to replace every other multi-layer perceptron (MLP) layers with sparse MoE layers. Each expert network is of the same architecture: $\mathrm{MLP}(x) = W_2\sigma_{\text{gelu}}\left(W_1 x\right)$, where $W_1 \in \mathbb{R}^{d_m \times d_f}$ and $W_2 \in \mathbb{R}^{d_f \times d_m}$ are learanble weights while σ_{gelu} is the non-linear activation layer (Hendrycks & Gimpel, 2016). It is worth noting that MoE is applied to multiple visual tokens, where each token could have different expert choice.

We also employ an auxiliary loss to encourage the load balancedness following (Shazeer et al., 2017) termed as $\mathcal{L}^{\text{lb}}$ to prevent the over-selection of few experts.

3.2 SPARSE MIXTURE OF EXPERTS FOR CONTRASTIVE LEARNING

To enforce the consistency of expert selection while not leaking image identity, we introduce a new regularization term called Overlapping based Gate Aligning Regularization (OGAR). In ViT, MoE

layer would choose experts for each token. The token sequence includes one classification token and multiple patch tokens. We then introduce how OGAR is applied for classification and patch tokens.

OGAR for classification tokens As the classification token is at the image level, enforcing the consistency can be easily realized by applying similarity constrain among classification tokens for augments of the same image. Formally, it is defined as

$$\mathcal{L}^G_{[\mathrm{CLS}]} = \mathcal{M}([\mathrm{CLS}]_1, [\mathrm{CLS}]_2, \{[\mathrm{CLS}]\}^-, \tau) \tag{6}$$

where $[\mathrm{CLS}]_1$ and $[\mathrm{CLS}]_2$ denote the feature of classification tokens from a pair of positive samples. $\{[\mathrm{CLS}]\}^-$ denotes that from negative samples. τ is the temperature, where we use the same value as $\mathcal{L}_{\mathrm{CL}}$. We employ the form of Moco V3 loss to enforce the consistency for preventing all the gate functions collapse to always outputting the same prediction.

OGAR for patch tokens Unlike classification tokens, different patches lack one-to-one correspondence as the patches are randomly sampled from different regions of the original image. Hence matching the patches is required before conducting the regularization. Previous studies reveal that the transformer can automatically learn object segmentation that aligns well with input in terms of the spatial location (Caron et al., 2021), which indicates the strong spatially correlation between input and features learned by CL. Inspired by this observation, we design a matching method based on the spatial location of the patches. Specifically, as shown in Figure 2, we first calculate the coordination information for each small patch, which is further utilized to calculate Intersection over Union (IoU). Afterward, each patch is paired with the patch from the other view with the highest IoU. If one patch is not overlapped with other patches, we would leave it unpaired. Only paired patches are utilized for calculating the loss. Formally, the proposed patch loss is defined as

$$\mathcal{L}^G_P = \frac{1}{N_p}\sum_{m=1}^{N_p}\mathcal{L}^G_{P_m}, \quad \mathcal{L}^G_{P_m} = \begin{cases} \mathcal{M}(p_m, p_n, \{p\}^-, G, \tau) & \text{if } \mathrm{IoU}_{mn} > \lambda \\ 0 & \text{otherwise.} \end{cases}, \quad n = \arg\max_{n'} \mathrm{IoU}_{mn'} \tag{7}$$

where N_p is the number of patches, p_m and p_n are the patch token features corresponding to mth and nth patches, respectively. IoU_{mn} is the IoU of patch m and n. nth patch has the largest overlap with mth patch. th is the IoU threshold. $\{p\}^-$ is the set of negative patches, which are composed by the patches from other images. $\mathcal{L}^G_{P_m}$ filters out patch pairs with a IoU smaller than λ.

Some previous works study a similar problem: enforcing the regional regularization of CL (Li et al., 2021; Wang et al., 2021), which also requires matching the local features. They match the features across two views based on the feature distance (e.g. cosine similarity). However, we empirically find this approach yield less significant improvement in our case. The intuition behind this is that the paired features in inter-mediate layers may lack strong feature similarity. The proposed matching method allows the existence of non-paired patches while the design of Wang et al. (2021) assumes all local features can be paired, which is prone to noise in learned features and also in general does not hold in practice.

We balance the two regularization terms with a convex combination controlled with a weight α ($0 < \alpha < 1$). Formally, the resulting OGAR is

$$\mathcal{L}^G = (1-\alpha)\mathcal{L}^G_{[\mathrm{CLS}]} + \alpha\mathcal{L}^G_P \tag{8}$$

The overall optimization target for CoMoE To sum up, the overall loss is

$$\mathcal{L} = \mathcal{L}_{\mathrm{CL}} + w_{\mathrm{lb}}\mathcal{L}^{\mathrm{lb}} + w_G\mathcal{L}^G \tag{9}$$

where w_{lb} and w_G are the scaling factor of the loading balancedness losses and OGAR, respectively. By employing OGAR on naive MoE+CL, the resultant CoMoE framework can efficiently scaling contrastive learning with MoE.

4 Experiment

4.1 Settings

Pre-training Our pre-training experiments are conducted on ImageNet-1K (Deng et al., 2009) following common practice (Chen et al., 2020a; He et al., 2020). For pre-training framework, we

Table 1: Network architecture comparison for four different architectures. CoMoE uses VMoE-S/16 as the backbone.

Model	Parameters	FLOPs	Throughput (images/s)
ResNet50	25M	4.1G	1226
ViT-S/16	22M	4.6G	936
VMoE-S/16	72M	4.6G	800
ViT-B/16	87M	17.6G	292

employ Moco v3 (Chen et al., 2021b), and we follow the same settings as Moco v3 on data augmentations and learning specification: 3-layer MLP projection head, temperature $\tau = 0.2$, momentum $m = 0.99$, random patch projection, cosine decay schedule (Loshchilov & Hutter, 2016), and 40-epoch warmup. For optimization, we employ AdamW (Loshchilov & Hutter, 2017) optimizer and a weight decay of 0.1. We employ linear scaling rule (Goyal et al., 2017) and search for the best base learning rate (lr) on 100-epoch results with grid of $\{1.5e^{-4}, 3.0e^{-4}, 5.0e^{-4}, 1.0e^{-3}\}$. The best searched lr is $5.0e^{-4} \times \mathrm{BatchSize}/256$. For model ablations, we employ a shorter schedule of 100 epochs with a relatively small batch size of 1024. When comparing with state-of-the-art methods, we scale up and employ 300 epochs with a batch size of 3072.

Linear probing Linear probing measures the quality of learned representations from pre-training. After self-supervised pre-training, we remove the MLP heads and train a classifier with the frozen backbone. Following Moco V3, we employ the SGD optimizer with a batch size of 4096 and weight decay of 0 for 90 epochs, with only random resized cropping and flipping augmentation. The lr is swept following common practice (Chen et al., 2021b; Zhou et al., 2021).

Semi-supervised and transfer few-shot learning Learning with few labels is an important application for contrastive learning, which pertains to both semi-supervised and transfer few-shot learning (Chen et al., 2020b; Tian et al., 2020b;b; Islam et al., 2021). Specifically, for semi-supervised learning, we consider consider 1% or 10% available labels (following the sampling in (Chen et al., 2020b)) of ImageNet. For transfer few-shot learning, we consider 4-shot and 10-shot setting for three datasets: CIFAR10 (Krizhevsky et al., 2009), Pet37 (Parkhi et al., 2012) and Food101 (Bossard et al., 2014).

For these two applications, we consider a two steps paradigm: The model is first pre-trained on the *pre-train* and then it is *supervised fine-tune* on the seed or few-shot dataset. For the *supervised fine-tune* step, we employ different settings for different tasks. As suggested in Tian et al. (2020b); Zhou et al. (2021), we train a linear classifier on frozen features for ImageNet 1% semi-supervised task and all transfer few-shot tasks. We optimize for 800 epochs with batchsize of 256 while other settings keeps the same as *linear probing*. For ImageNet 10% semi-supervised task, we follow Chen et al. (2020b); Zhou et al. (2021) fine-tuning from the first layer of the MLP head. The epochs number is set as 200 while the lr are searched with grid of $\{1e^{-5}, 3e^{-5}, 1e^{-4}, 3e^{-4}\}$.

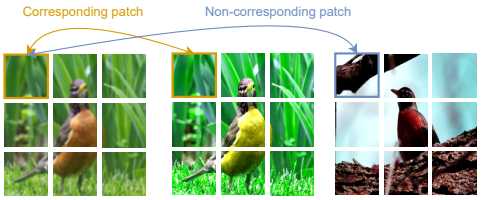
Hyper-parameters for Mixture-of-Experts Model and loss For MoE network, we by default employ 16 expert candidates ($n_e = 16$) and always activate 2 of them ($k = 2$). For each expert network, we choose $d_f = 2d_m$ instead of $d_f = 4d_m$ in Chen et al. (2021b) to keep the computational cost of activating 2 experts the same as that in ViT. The employed model is VMoE-S/16, as shown in table 1, its FLOPs and thoughput are comparable to ViT-S/16 and Resnet-50. For the employed loss terms, we employ $\lambda = 0.2$, $\alpha = 0.3$, $w_{\text{lb}} = 0.01$ and $w_G = 0.001$, which are searched on 100-epoch training.

Computation Framework Our implementation is based on Pytorch (Paszke et al., 2019) and FastMoE (He et al., 2021a) library. All experiments are conducted on 32 Nvidia V100 GPUs.

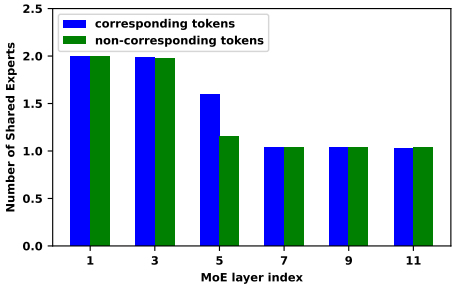
4.2 Naive Combination of MoE and CL Does Not Work

In this section, we look into the "cross-view instability" issue of directly plugging MoE to CL and show how the proposed regularization address this problem.

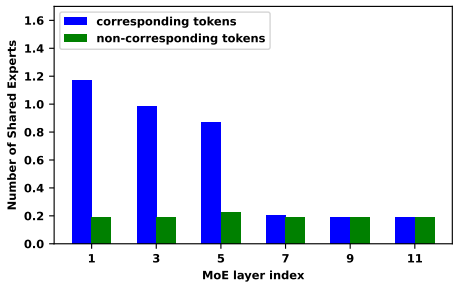
The routing is inconsistent To check the consistency of the expert decision, as shown in Figure 3a, we exclude random cropping and flipping from data augmentations to ensure we can locate the different views of the same patches: they are always in the same position in this way. Further, we define these patches with the same content as *corresponding tokens* while defining the tokens from



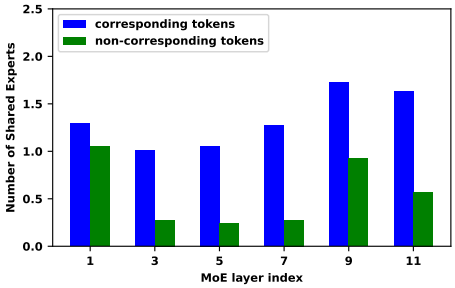
(a) corresponding and non-corresponding patches



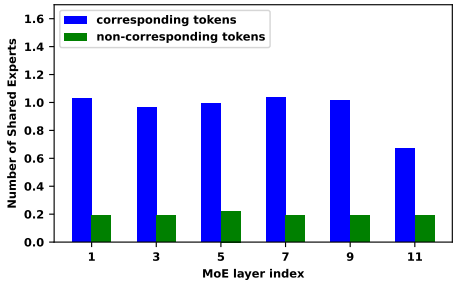
(b) [CLS] tokens w/o regu



(c) Patch tokens w/o regu



(d) [CLS] tokens w/ regu



(e) Patch tokens w/ regu

Figure 3: (a) Illustration for the definition of corresponding and non-corresponding patches. The rest four figures compare the average number of shared experts for $G(x)$ between corresponding and non-corresponding tokens. (a)(c) shows the number of shared experts for classification tokens while (b)(e) represents the number of shared experts for patch tokens. All of them are measured across different layers. The "w/o regu" in (a)(b) denotes they are from the naive combination CL and MoE model. In contrast, (b)(e) is from the proposed CoMoE. The x-axis of the last four figures is the index of the MoE layer in VMoE.

other images as the *non-corresponding tokens*. Then, we calculate the average number of shared experts (the number of experts selected by both tokens in the pair) for *corresponding tokens* and *non-corresponding tokens* and make a comparison.

As shown in Figure 4a, for classification tokens of the naive combination, the gating function always selects the similar number of shared experts between *corresponding* and *non-corresponding patches*, which means the difference between corresponding patches and non-corresponding patches can hardly be distinguished. For the patch tokens, as presented in Figure 4b, the boundary between *corresponding* and *non-corresponding patches* get blurred in the deep layers. This would change the standard contrasting shared weight backbone fashion of CL to contrasting (partially) non-shared weight contrastive learning.[1]

Inconsistent routing leads performance dropping Unfortunately, the proof-of-concept experiments verify that performance of (partially) non-shared weight contrastive learning can drop. Specifically, we designed a special network called sep-ViT, which has the same backbone architecture as MoE with 2 expert candidates. For routing, we would activate different experts for different

[1]Contrasting with a moving average network can be regarded as sharing weight as the moving average would converge to the online value when training stabilize.

Table 2: Linear probing (denote as linear) and 1% imagenet semi-supervised (denote as 1%) performance comparison for pre-training and evaluation on ImageNet. All the reported accuracy is top 1 accuracy (%). expert 0/1 for sep-ViT-S denote the two different paths of sep-ViT-S.

Method	Model	Linear	1%
Moco v3	ViT-S/16	69.7	53.5
Moco v3	sep-ViT-S/16 (expert 0)	68.4	48.5
Moco v3	sep-ViT-S/16 (expert 1)	68.5	48.7
Moco v3	V-MoE-S/16	69.9	54.1
CoMoE (Ours)	V-MoE-S/16	**70.7**	**55.8**

Table 3: Comparison with State-of-The-Art methods in terms of linear probing (denote as Linear), 1% and 10% semi-supervised performance (denote as 1% and 10%, respectively). All the reported accuracy is top 1 accuracy (%). The SD denotes self-distillation.

Match method	Model	Linear	1%	10%
SimCLR v2 (Chen et al., 2020b)	Resnet50	71.7	57.9	68.1
SimCLR v2 + SD (Chen et al., 2020b)	Resnet50	71.7	60.0	70.5
Moco v3 (Chen et al., 2021b)	ViT-S/16	73.4	59.4	72.2
CoMoE (ours)	V-MoE-S/16	**74.1**	**62.2**	**73.0**

branches. In this way, these two branches would not share weight in the expert network. The result is illustrated in Table 2, the sep-ViT-S (expert 0) decrease the performance by 0.8% and 5% for linear probing and 1% semi-supervised performance compared to the baseline, respectively, indicating that (partially) non-shared weight can hurt the performance for CL (especially for the semi-supervised performance).

The proposed CoMoE improves both consistency and performance After employing the proposed classification alignment and OGAR, as shown in Figure 4c and 4d, the proposed CoMoE successfully increase the number of shared experts for *corresponding tokens* while reducing or keeping the number of the shared experts for *non-corresponding tokens*. Also, as shown in Table 2, in contrast to the naive combination of CL and MoE that only improves the baseline Moco V3 by a small margin of 0.2% and 0.6% in terms of linear probing and 1% semi-supervised performance, the proposed CoMoE increase this margin to 1.0% and 2.3%, demonstrating the effectiveness of the proposed method.

4.3 Comparison with State-of-The-Art Methods

In this section, we compare the proposed CoMoE with state-of-the-art methods. For a fair comparison, we employ a longer training schedule of 300 epochs following Chen et al. (2021b); Caron et al. (2021); Zhou et al. (2021).

CoMoE yield better in-domain performance As shown in Table 3, the proposed CoMoE achieves highest performance in terms of Linear probing, 1% and 10% semi-supervised learning. Remarkably, compared to Moco v3 on ViT-S/16, the proposed CoMoE significantly improves the 1% semi-

Table 4: Transfer few-shot performance comparison across different datasets between MocoV3 and the proposed CoMoE with ViT-S/16. 4-shot and 10-shot denote 4 and 10 samples available for each class for downstream tasks, respectively. All the reported accuracy is top 1 accuracy (%). All the experiments are conducted on five different folders. The average and variance are reported.

Dataset	Method	4-shot	10-shot
CIFAR10	Moco V3	74.5±2.2	82.2±1.2
	CoMoE	75.7±1.0	82.6±1.2
Pet37	Moco V3	72.9±0.9	81.3±1.1
	CoMoE	76.5±1.1	83.7±1.0
Food101	Moco V3	36.3±0.7	50.0±0.8
	CoMoE	37.7±0.5	50.9±0.8

Table 5: Comparison between different hyper-parameters settings of the proposed CoMoE. Linear probing (denote as linear) and 1% imagenet semi-supervised (denote as 1%) performance are reported. All the reported accuracy is top 1 accuracy (%). The first row denotes the employed hyper-parameter setting. Error bar is calculated by running 3 times with different random seeds.

w_G	α	λ	Linear	1%
0.001	0.3	0.2	70.7±0.07	55.8±0.25
0.001	0.0	0.2	70.6±0.13	55.4±0.14
0.01	0.3	0.2	70.5	55.8
0.001	0.3	0.1	70.6	55.9

supervised performance by 2.8%. Meanwhile, there is also a non-trivial improvement on Linear Evaluation and 10% semi-supervised performance by 0.6% and 0.8%, respectively. Since the Moco V3 and CoMoE share the same CL framework, this demonstrate the effectiveness of MoE framework and the proposed regularization. The large improvement on semi-supervised performance also matches the observation at Chen et al. (2020b) that large capacity helps more for few-shot learning.

CoMoE yields better transfer few-shot performance We then study if the strong in-domain few-shot performance can transfer to downstream datasets. As demonstrated in Table 4, the proposed CoMoE also yields an consistent improvement of [1.2%,0.4%], [3.6%,2.4%] and [1.4%,0.9%] for CIFAR10, Pet37 and Food101, respectively, in terms of [4-shot, 10-shot] performance, demonstrating the proposed CoMoE can also significantly improve the downstream few-shot performance. Note that the few-shot performance is evaluated on five randomly sampled folders. To further verify the significance of the improvement, we compare the performance on each split at Appendix A.2 and find that the proposed method yields a statistically significant gain over Moco V3.

4.4 Ablation Studies

OGAR loss for patch tokens matters As shown in Table 5, when remove OGAR loss for patch tokens by setting $\alpha = 0$, the linear evaluation and 1% semi-supervised performance would drop by [0.1%, 0.4%], which demonstrate the effectiveness of the proposed OGAR loss for patch tokens. Other hyper-parameter change like $w_G = 0.01$ and $\lambda = 0.1$ only marginally change the performance.

OGAR loss ablation In Table 6, we ablation study the proposed OGAR loss. When switching from the overlap-based matching method to the Feature Similarity-based Matching method (FSM), the linear probing and 1% semi-supervised performance would both incur a drop of 0.2%. When discarding the negative samples for OGAR loss and only enforcing consistency as in Grill et al. (2020), we observe the gating function tent to choose the same experts for all samples even though we have employed the loading balance loss. Meanwhile, the performance would largely decrease, showing that negative samples are necessary for OGAR.

Table 6: OGAR loss ablation regarding different matching methods and whether to employ negative samples. FSM denotes the Feature Similarity-based Matching method employed in Li et al. (2021); Wang et al. (2021).

Matching Method	Negative samples	Linear	1%
FSM	✓	70.5±0.07	55.6±0.37
Overlap		70.2	54.2
Overlap	✓	70.7±0.07	55.8±0.25

5 Conclusion

In this work, we study an efficient way of scaling contrastive learning with sparse Mixture of Experts. We start from naively plugging in the MoE to CL and observe that the naive combination tends to route different views of the same image to different subsets of experts, thus breaking invariant feature learning and hurting the performance of down-stream tasks. To tackle this problem, we propose a novel regularization framework to promote consistency of experts selection on the same (or overlapped) image tokens while encouraging diversity of the experts selection for different images. Extensive evaluations on multiple downstream tasks demonstrate the proposed framework, CoMoE, effectively improve the routing consistency and the overall performance of downstream tasks without increasing the computation cost.

REFERENCES

Karim Ahmed, Mohammad Haris Baig, and Lorenzo Torresani. Network of experts for large-scale image categorization. In *European Conference on Computer Vision*, pp. 516–532. Springer, 2016.

Dosovitskiy Alexey, Philipp Fischer, Jost Tobias, Martin Riedmiller Springenberg, and Thomas Brox. Discriminative, unsupervised feature learning with exemplar convolutional, neural networks. *IEEE TPAMI*, 38(9):1734–1747, 2016.

Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*, 2014.

Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2229–2238, 2019.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

Ke Chen, Lei Xu, and Huisheng Chi. Improved learning algorithms for mixture of experts in multiclass classification. *Neural networks*, 12(9):1229–1252, 1999.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020b.

Ting Chen, Calvin Luo, and Lala Li. Intriguing properties of contrastive losses. *Advances in Neural Information Processing Systems*, 34, 2021a.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, 2021.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020c.

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9640–9649, 2021b.

Rumen Dangovski, Li Jing, Charlotte Loh, Seungwook Han, Akash Srivastava, Brian Cheung, Pulkit Agrawal, and Marin Soljačić. Equivariant contrastive learning. *arXiv preprint arXiv:2111.00899*, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. Glam: Efficient scaling of language models with mixture-of-experts. *arXiv preprint arXiv:2112.06905*, 2021.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *arXiv preprint arXiv:2101.03961*, 2021.

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33:21271–21284, 2020.

Sam Gross, Marc'Aurelio Ranzato, and Arthur Szlam. Hard mixtures of experts for large scale weakly supervised vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6865–6873, 2017.

Jiaao He, Jiezhong Qiu, Aohan Zeng, Zhilin Yang, Jidong Zhai, and Jie Tang. Fastmoe: A fast mixture-of-expert training system. *arXiv preprint arXiv:2103.13262*, 2021a.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*, 2021b.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Ashraful Islam, Chun-Fu Richard Chen, Rameswar Panda, Leonid Karlinsky, Richard Radke, and Rogerio Feris. A broad study on the transferability of visual representations with contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8845–8855, 2021.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

Michael I Jordan and Robert A Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214, 1994.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*, 2020.

Mike Lewis, Shruti Bhosale, Tim Dettmers, Naman Goyal, and Luke Zettlemoyer. Base layers: Simplifying training of large, sparse models. In *International Conference on Machine Learning*, pp. 6265–6274. PMLR, 2021.

Chunyuan Li, Jianwei Yang, Pengchuan Zhang, Mei Gao, Bin Xiao, Xiyang Dai, Lu Yuan, and Jianfeng Gao. Efficient self-supervised vision transformers for representation learning. *arXiv preprint arXiv:2106.09785*, 2021.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Jian Meng, Li Yang, Jinwoo Shin, Deliang Fan, and Jae-sun Seo. Contrastive dual gating: Learning sparse features with contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12257–12265, 2022.

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6707–6717, 2020.

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pp. 69–84. Springer, 2016.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pp. 3498–3505. IEEE, 2012.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Svetlana Pavlitskaya, Christian Hubschneider, Michael Weber, Ruby Moritz, Fabian Huger, Peter Schlicht, and Marius Zollner. Using mixture of expert models to gain insights into semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 342–343, 2020.

Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Scaling vision with sparse mixture of experts. *Advances in Neural Information Processing Systems*, 34, 2021.

Stephen Roller, Sainbayar Sukhbaatar, Jason Weston, et al. Hash layers for large sparse models. *Advances in Neural Information Processing Systems*, 34, 2021.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *Advances in Neural Information Processing Systems*, 33:6827–6839, 2020a.

Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? In *European Conference on Computer Vision*, pp. 266–282. Springer, 2020b.

Trieu H Trinh, Minh-Thang Luong, and Quoc V Le. Selfie: Self-supervised pretraining for image embedding. *arXiv preprint arXiv:1906.02940*, 2019.

Tsung Wei Tsai, Chongxuan Li, and Jun Zhu. Mice: Mixture of contrastive experts for unsupervised image clustering. In *International Conference on Learning Representations*, 2020.

Yao-Hung Hubert Tsai, Paul Pu Liang, Amir Zadeh, Louis-Philippe Morency, and Ruslan Salakhutdinov. Learning factorized multimodal representations. *arXiv preprint arXiv:1806.06176*, 2018.

Xin Wang, Fisher Yu, Lisa Dunlap, Yi-An Ma, Ruth Wang, Azalia Mirhoseini, Trevor Darrell, and Joseph E Gonzalez. Deep mixture of experts via shallow embedding. In *Uncertainty in artificial intelligence*, pp. 552–562. PMLR, 2020.

Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3024–3033, 2021.

Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3733–3742, 2018.

Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. *arXiv preprint arXiv:2111.09886*, 2021.

Fuzhao Xue, Ziji Shi, Futao Wei, Yuxuan Lou, Yong Liu, and Yang You. Go wider instead of deeper. *arXiv preprint arXiv:2107.11817*, 2021.

Brandon Yang, Gabriel Bender, Quoc V Le, and Jiquan Ngiam. Condconv: Conditionally parameterized convolutions for efficient inference. *Advances in Neural Information Processing Systems*, 32, 2019.

Seniha Esen Yuksel, Joseph N Wilson, and Paul D Gader. Twenty years of mixture of experts. *IEEE transactions on neural networks and learning systems*, 23(8):1177–1193, 2012.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pp. 12310–12320. PMLR, 2021.

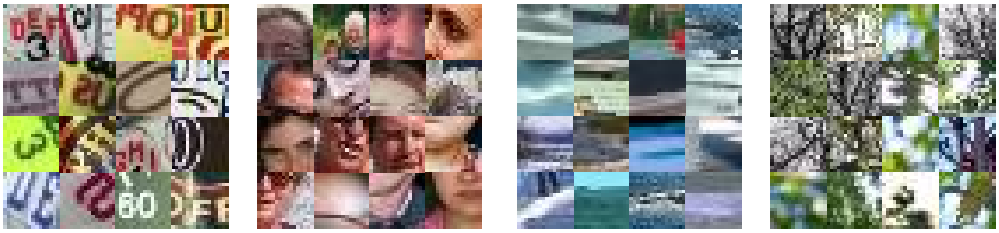
Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

Barret Zoph, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer, and William Fedus. Designing effective sparse expert models. *arXiv preprint arXiv:2202.08906*, 2022.

A Appendix

This appendix contains the following details that we could not include in the main paper due to space restrictions.

A.1 Visualize the Routing of Experts



(a) Expert 1 (Characters) (b) Expert 2 (Faces) (c) Expert 4 (Pool/Sea) (d) Expert 5 (Forest/Tree)

Figure 4: Visualization of the patch tokens routed to different experts in the 7th layer of CoMoE on ImageNet. The patches with different patterns are routed to different experts.

The visualization of the routing can be found in the Figure 4. We find that the patches routed to different tokens show different patterns. For example, the patches of [Characters, Faces, Pool/Sea, Forest/Tree] are routed to Expert [1, 2, 4, 5], respectively.

A.2 Detailed Few-shot Transfer Comparison

We further extend the few-shot transfer comparison on experiments with overlapping variance and present the performance for each split in Table 7. We can find that the gain of CoMoE above Moco V3 on [CIFAR10 4-shot, CIFAR10 10-shot, Food101 10-shot] are [1.6$\pm$1.4, 0.3$\pm$0.2, 1.0$\pm$0.1]. The consistently improvement over Moco v3 demonstrates the effectiveness of the proposed CoMoE.

Table 7: Transfer few-shot performance comparison between MocoV3 and the proposed CoMoE with ViT-S/16 on different splits. All the reported accuracy is top 1 accuracy (%). (a) (b) and (c) corresponds CIFAR10 4-shot, CIFAR10 10-shot and Food101 10-shot, respectively. The difference denotes the subtraction of Moco V3 from CoMoE.

(a) CIFAR10 4-shot

Method	Split 1	Split 2	Split 3	Split 4	Split 5	Mean
Moco V3	77.8	75.0	72.0	72.2	74.6	74.5±2.2
CoMoE	77.6	75.6	74.7	75.2	75.4	75.7±1.0
Difference	-0.2	0.5	2.7	3.0	0.8	1.4±1.3

(b) CIFAR10 10-shot

Method	Split 1	Split 2	Split 3	Split 4	Split 5	Mean
Moco V3	80.4	83.3	82.7	82.2	82.3	82.2±1.0
CoMoE	81.0	83.8	83.2	82.0	82.8	82.6±1.0
Difference	0.6	0.5	0.5	-0.2	0.5	0.4±0.3

(c) Food101 10-shot

Method	Split 1	Split 2	Split 3	Split 4	Split 5	Mean
Moco V3	49.1	49.2	50.1	50.0	51.4	50.0±0.8
CoMoE	49.9	50.3	51.1	51.1	52.3	50.9±0.8
Difference	0.8	1.1	1.0	1.1	0.9	1.0±0.1

A.3 With Other Backbones

In this section, we explore the performance of the proposed method on a different backbone. As shown in Table 8, CoMoE with V-MoE-B/16 surpasses the Moco V3 with ViT-B/16 by 1% in terms of the 1% few-shot performance while leading to a small drop of 0.4% on linear evaluation performance.

Table 8: Illustration of the performance in terms of linear probing (denote as Linear), 1% semi-supervised performance (denote as 1%). All the reported accuracy is top 1 accuracy (%).

Match method	Model	Linear	1%
Moco v3 (Chen et al., 2021b)	ViT-B/16	76.7	63.9
CoMoE (ours)	V-MoE-B/16	76.3	64.9