CLARE: SCALABLE CLASS-INCREMENTAL CONTINUAL LEARNING VIA A SPARSITY-BASED FRAMEWORK

Anonymous authors
Paper under double-blind review

ABSTRACT

The primary challenge in continual learning is navigating the plasticity-stability dilemma to balance the acquisition of new knowledge with the retention of old. While leveraging pretrained models has significantly advanced continual learning, existing methods exhibit a scalability bottleneck on long task sequences, suffering from performance degradation due to parameter interference and loss of plasticity. In this work, inspired by evidence that sparse fine-tuning achieves performance comparable to full fine-tuning, we introduce a novel sparsity-driven continual learning framework. Our continual learning method termed CLARE operates in two stages: it first identifies a sparse, task-critical parameter mask via a sparsity-inducing objective, then performs mask-constrained fine-tuning. In addition, to further reduce interference, we incorporate a gradual forgetting mechanism that resets a tiny fraction of previously accumulated parameters after learning each new task. Furthermore, to address the lack of benchmark datasets for long-sequence continual learning, we curate ImageNet-CIL-1K, a challenging long-sequence dataset with 1,069,563 images and 1,000 classes. Extensive experiments demonstrate the scalability of CLARE. On ImageNet-CIL-1K with 100 tasks, CLARE outperforms strong baselines such as APER and MagMax by 4-6% in overall test accuracy, and leads EASE by over 10%, establishing a new state of the art for long-sequence continual learning.

1 INTRODUCTION

The core challenge of continual learning (CL) and class incremental learning (CIL) lies in achieving a balance between the capacity to learn new diverse tasks (*learning plasticity*) and the ability to retain previously learned knowledge without catastrophic forgetting (*memory stability*). Traditional CIL methods can often be categorized into three main paradigms: regularization-based methods (Kirkpatrick et al., 2017; Li & Hoiem, 2017), replay-based methods (Lopez-Paz & Ranzato, 2017), and optimization-based methods (Farajtabar et al., 2020). Recent advancements leverage strong pretrained models (PTM) to further improve performance instead of training models from scratch, as pretrained models encapsulate rich prior knowledge. In particular, building adapter-based (Zhou et al., 2024; Yu et al., 2024a; Gao et al., 2025) or model merging based (Marczak et al., 2024; Gao et al., 2025) continual learning models on top of a pretrained backbone represents two prominent directions.

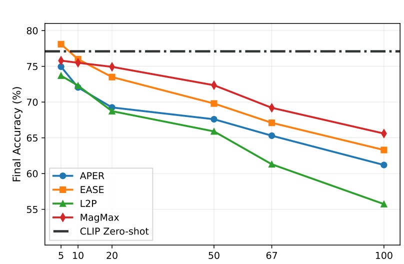

Figure 1: Overall accuracy of continual learning methods on ImageNet-R split into varying numbers of tasks.

Although recent PTM-based CIL methods have shown promising performance on short task sequences (e.g., 10 tasks), scaling these methods to longer task sequences typically means substantial performance sacrifice. To verify this, we present a pilot study on up to 100 tasks based on the ImageNet-R dataset (Hendrycks et al., 2021a) of four representative continual learning baselines, including APER (Zhou et al., 2025), EASE (Zhou et al., 2024), L2P (Wang et al., 2022b), and MagMax (Marczak et al., 2024). The results in Figure 1 shows that existing methods lag behind CLIP zero-shot (Radford et al., 2021), a gap that widens with the number of tasks. This decline stems primarily from an imbalance between

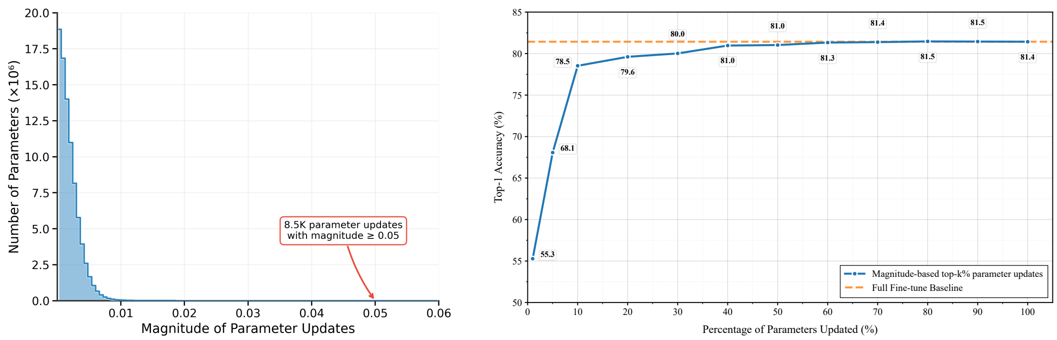

Figure 2: Sparse parameter update analysis. **Left:** Long-tail distribution of parameter update magnitudes shows most parameters experience tiny updates (< 0.01), while only 8.5K parameters have updates ≥ 0.05. **Right:** Sparse parameter updates achieve performance close to full fine-tuning on ImageNet-R using ImageNet-1K pretrained ViT-B/16.

interference and plasticity. As the task sequence lengthens, effective new task learning causes catastrophic forgetting of earlier knowledge as new updates overwrite or conflict with parameters crucial for previous tasks. However, effective earlier knowledge preservation restricts a model's capacity to integrate new information, resulting in progressively poorer performance on new tasks.

In this paper, we hypothesize that strategically learning a small number of parameters for each task can already maintain sufficient plasticity while dramatically reducing the likelihood of destructive interference across tasks. This hypothesis is supported by existing literature on learning sparse neural networks (Wen et al., 2016; Louizos et al., 2018; Ma et al., 2019) as well as empirical evidence showing that the magnitude of parameter updates during fine-tuning follows a long-tailed distribution (Figure 2 (left)), with substantial updates being confined to a tiny subset of parameters. More importantly, it is only necessary to update a small proportion of the model parameters to achieve competitive task-specific performance, as illustrated in Figure 2 (right). On the basis of this insight, we propose a sparsity-driven continual learning framework that learns a sparse subset of parameters for each task, enabling effective scaling to extended sequences while balancing the plasticity-stability trade-off in continual learning models.

Our sparsity-driven framework manages parameter allocation across task sequences through a two-stage learning process. Starting from a pretrained base model, we first identify task-critical parameters by optimizing a sparsity-inducing objective, which produces a binary mask identifying the most relevant parameters for the task. We then perform mask-constrained fine-tuning, updating only these relevant parameters while keeping the remainder frozen. This enables the model to achieve promising performance by updating only a sparse subset of the total parameters, thereby facilitating targeted knowledge acquisition with minimal interference and preserved plasticity. In practice, the parameters learned for new tasks are incrementally fused into the base model via simple accumulation for computational efficiency. To further mitigate interference arising from repeated parameter use, that is, the same parameter may be updated across multiple tasks, we introduce a gradual forgetting mechanism. This mechanism randomly resets a tiny portion of the accumulated updates to zero when parameter reuse exceeds a predefined saturation threshold, effectively reducing interference and enhancing stability during continual learning.

In terms of performance evaluation, existing continual learning benchmark datasets suffer from a limited number of classes. For instance, CIFAR-100 is a foundation dataset for the field but presents a deficiency for long task sequences due to its class count (100 classes). Dividing CIFAR-100 into 100 tasks results in single-class learning episodes that do not possess the complexity of realistic continual learning tasks. To fill this critical gap in evaluation protocols, we introduce ImageNet-CIL-1K, a challenging long-sequence benchmark dataset comprising 1,000 classes curated from ImageNet-21K, with 1069 images per class on average. This dataset is a comprehensive testbed for long-sequence continual learning methods.

We evaluate the performance of CLARE through extensive experiments. The results demonstrate that our method achieves significant improvements over strong continual learning baselines when using a similar number of trainable parameters. Specifically, on our ImageNet-CIL-1K dataset with 100 tasks and 10 classes per task, our method achieves remarkable gains of over 6% and 4% in average test accuracy over MagMax (Marczak et al., 2024) and APER (Zhou et al., 2025), respectively. Compared to the recently proposed MoAL (Gao et al., 2025), CLARE improves average test accuracy by over 10%. We also perform evaluations on shorter task sequences from the ImageNet-CIL-1K dataset, including 50 and 75 tasks. The results consistently surpass previous CIL methods by a large margin. The above findings indicate that our method can effectively perform long-sequence continual learning.

To further explore the generalizability of CLARE, we also perform extensive evaluations on standard CIL datasets, including CIFAR-100, ImageNet-R, ImageNet-A, and OmniBenchmark, using the same evaluation protocols as in previous work. Our method maintains superior performance with respect to the existing baselines. For example, compared to APER (Zhou et al., 2025), our method improves the average test accuracy across the four benchmarks by 6.5%.

2 METHOD

Problem Definition. Consider a neural network $\mathcal{F}_\phi : \mathcal{X} \to \mathcal{Y}$ parameterized by $\{\theta, \phi\}$, where θ represents the backbone parameters and ϕ denotes the classifier parameters. The backbone is initialized from pretrained weights θ_{base} and fine-tuned on a sequence of T tasks while the classifier is trained from scratch. At step $k \in \{1, \dots, T\}$, the model receives dataset $\mathcal{S}_k = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_k}$ where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \mathcal{C}_k$ with $|\mathcal{C}_k| = m_k$ classes, and learn task k by incrementally updating the backbone and classifier parameters using $\mathcal{S}_k$. This paper focuses on class incremental learning (CIL), where the class spaces are disjoint: $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$ for $i \neq j$. Under the *exemplar-free* constraint, the previous data $\{\mathcal{S}_1, \dots, \mathcal{S}_{k-1}\}$ becomes inaccessible when learning the task k. The model must generalize to the cumulative label space $\mathcal{C}^{(k)} \left(= \bigcup_{i=1}^{k} \mathcal{C}_i\right)$ containing $\sum_{i=1}^{k} m_i$ total classes.

We are particularly interested in the long task sequence regime where T is relatively large (e.g., $T > 20$), since this is an unexplored problem in previous works. In this context, catastrophic forgetting is exacerbated, and there exist new challenges related to model capacity and parameter interference that are insignificant in shorter sequences.

2.1 TWO-STAGE SPARSITY-DRIVEN LEARNING

To ensure the simplicity and efficiency of the proposed method during inference (Ke et al., 2024; Marczak et al., 2024), we start with a simple model merging paradigm. Specifically, a base backbone model θ_{base} is individually fine-tuned on every task in a sequence to obtain task-specific models $\{\theta_k\}_{k=1}^{T}$, which are subsequently merged to obtain the model for the entire sequence. For incoming task k with dataset $\mathcal{S}_k$, we define the parameter update as

$$\Delta\theta_k = \theta_k - \theta_{\text{base}}, \tag{1}$$

where $\Delta\theta_k$ encodes task-specific knowledge. The final model is obtained by adding $\Delta\theta_k$ for all learned tasks:

$$\theta_{\text{merged}} = \theta_{\text{base}} + \Delta\theta_{\text{accu}}, \quad \text{where} \quad \Delta\theta_{\text{accu}} = \sum_{k=1}^{T} \Delta\theta_k. \tag{2}$$

This learning algorithm is general and can be coupled with existing classifiers for prediction, such as prototype-based approaches (Zhou et al., 2025). In this model merging paradigm, parameter interference occurs when parameter updates of different tasks correlate with each other. Parameter interference disrupts the ability of θ_{merged} to maintain task-specific knowledge, leading to catastrophic forgetting as knowledge of old tasks is damaged. Given a long sequence of tasks, the probability of parameter interference increases rapidly, because each task k could conflict or correlate with the remaining $T - 1$ tasks. Meanwhile, it is crucial to ensure sufficient model plasticity to adequately grasp new tasks. Thus achieving a balance between parameter interference and model plasticity is the key for long sequence CIL.

To tackle this challenge, we propose to explicitly learn a small subset of parameters that are most relevant for each task, yielding sparse and representative parameter updates. This mitigates parameter interference in θ_{merged} while maintaining adequate model plasticity. Specifically, the sparse subset of most relevant parameters is discovered via including an L_1 regularization term, which facilitates sparse signal recovery (Donoho & Stark, 1989; Donoho & Logan, 1992), in addition to the cross-entropy term in the training loss. Since optimizing such a joint objective could lead to suboptimal classification performance, we introduce a two-stage fine-tuning procedure that separates mask discovery from task learning.

Stage 1: Sparse Mask Discovery We first learn which parameters are most relevant for task k by minimizing

$$\mathcal{L}_{\text{mask}} = \underbrace{\mathbb{E}_{(x,y)\sim\mathcal{S}_k}\left[\ell(f(x;\tilde{\theta}_k),y)\right]}_{\text{Cross-entropy loss}} + \lambda\|\tilde{\theta}_k - \theta_{\text{base}}\|_1, \tag{3}$$

where $\tilde{\theta}_k$ is the unknown and λ is a hyperparameter that balances the two terms in the above loss. The L_1 regularization encourages sparsity in $\Delta\tilde{\theta}_k (= \tilde{\theta}_k - \theta_{\text{base}})$, naturally identifying the most relevant parameters for task k. Once the above loss minimization is complete, we compute a sparse binary mask $M_k \in \{0,1\}^{|\theta|}$ with a sparsity ratio ρ denoting the percentage of zero entries in M_k. Specifically, we calculate the ρ percentile of all $|\Delta\tilde{\theta}_k|$ values as a threshold to generate the sparse mask M_k. In our experiments, ρ is set to 85-96%.

Stage 2: Mask-Constrained Task Learning Using the discovered mask M_k, we perform standard fine-tuning with cross-entropy loss only:

$$\theta_k = \operatorname*{argmin}_{\tilde{\theta}_k} \mathbb{E}_{(x,y)\sim\mathcal{S}_k}\left[\ell(f(x;\tilde{\theta}_k),y)\right] \quad \text{s.t.} \quad \text{grad}[\tilde{\theta}_k^{(i)}] = 0 \text{ if } M_k^{(i)} = 0, \tag{4}$$

where only the sparse subset of parameters chosen by the mask M_k receives gradient updates.

This two-stage process isolates mask learning from task learning, eliminating potential interference between the cross-entropy and L_1 regularization terms in the joint objective (Eq. 3). And the second stage focuses solely on task learning (Eq. 4), thus improving model plasticity. Note that Stage 2 incurs little computational overhead by only updating a sparse subset of model parameters. Comparison of training time is provided in the Appendix A.3.

Noise-Enhanced Robustness To further enhance robustness against parameter interference during model merging, we inject Gaussian noise during the forward pass of Stage 2 training:

$$\tilde{\theta}_k^{f^{(i)}} = \begin{cases} \theta_{base}^{(i)} + \mathcal{N}(0,\sigma^2), & \text{if } M_k^{(i)} = 0 \text{ (frozen)}, \\ \tilde{\theta}_k^{(i)}, & \text{if } M_k^{(i)} = 1 \text{ (active)}, \end{cases} \tag{5}$$

where $\tilde{\theta}_k^{f^{(i)}}$ stands for the parameter values used during the forward pass. This noise is applied to frozen parameters only during forward propagation, simulating the sum of $\Delta\theta_k$ for all other tasks in Eq. 2. Such noise adaptation improves the model's resilience to interference without affecting active parameter learning.

2.2 Gradual Forgetting

Our sparsity-driven learning via masking encourages a minimum number of parameter updates during task learning. Nevertheless, a subset of parameters is inevitably updated by multiple tasks, creating interference hotspots that can degrade performance over a long sequence. For a large number of tasks (T), the expected parameter usage may exceed the total number of parameters (i.e, $(1-\rho)T > 100\%$), giving rise to each parameter on average being updated multiple times and thus causing parameter interference. To this end, we introduce a gradual random forgetting mechanism because learning new tasks without forgetting cannot deliver optimal performance any more once the model capacity has been reached. In our mechanism, we randomly set the parameters in θ_{accu} to zero with a probability of $1/F$, where F is a hyperparameter that represents the saturation point, every time a new $\Delta\theta_k$ ($k > F$) is learned and added to existing θ_{accu}. This mechanism ensures that the average number of times each parameter is reused ceases to increase and is maintained at $(1-\rho)F$. The random forgetting mask $\mathcal{F}$ is applied as $\Delta\theta_{\text{acc}} \leftarrow \mathcal{F} \odot \Delta\theta_{\text{acc}}$, where $\mathcal{F} \sim \text{Bernoulli}(1 - 1/F)$. This process is consistent with the Ebbinghaus forgetting curve, where the fraction of memory that can be retained over time follows an exponential decay curve (Ebbinghaus, 1964).

2.3 Inference

Our sparsity-driven learning framework yields merged backbone parameters θ_{merge} alongside a collection of task-specific classifiers. Since the task identity, which is required to select the correct classifier, is unknown at inference time, we employ a Mixture-of-Experts (MoE) classifier equipped with an automatic routing mechanism to predict the task identity and direct the input to the selected expert. To enhance robustness against routing errors, we construct our router using an ensemble strategy following (Yu et al., 2024a; Gao et al., 2025). Details can be found in the Appendix A.1.

2.4 Discussions

There exist important differences between our two-stage sparsity-driven task learning and traditional regularization-based or sparse continual learning methods (Kirkpatrick et al., 2017; Zenke et al., 2017; Aljundi et al., 2018; Wang et al., 2022a). First, the learning processes are different. Our method separates mask learning from task learning and achieves optimal task learning over a sparse mask, while traditional methods attempt to learn new tasks and preserve previous knowledge simultaneously, resulting in suboptimal solutions for both objectives. Second, parameter selection criteria are different. Traditional methods use static, instantaneous heuristics like parameter magnitude or single-time gradient scores. In contrast, our sparse mask selects parameters by their accumulated updates during Stage-1 training, capturing their full contribution to learning.

3 Benchmark Datasets and Architectures

3.1 New Benchmark Dataset ImageNet-CIL-1K

Dataset Construction. Dividing existing class incremental learning (CIL) benchmark datasets, such as CIFAR-100 (Krizhevsky et al., 2009) and ImageNet-R (Hendrycks et al., 2021a), into long sequences of tasks would result in trivial or oversimplified tasks due to their limited total number of classes. For instance, dividing CIFAR-100 and ImageNet-R into 100 tasks yields only 1 and 2 classes per task, respectively. To this end, we construct ImageNet-CIL-1K, a challenging benchmark dataset for long-sequence continual learning that maintains task difficulty even when partitioned into many tasks. ImageNet-CIL-1K is derived from ImageNet-21K-P (Ridnik et al., 2021), a subset of ImageNet-21K (Deng et al., 2009) with 12,358,688 images from 11,221 classes after the exclusion of classes with fewer than 500 images. We exclude all classes present in ImageNet-1K from this subset to avoid data leakage when models pretrained on ImageNet-1K are used. We further remove images corresponding to non-leaf nodes in the WordNet hierarchy (Miller, 1995) to prevent taxonomic overlaps (e.g., eliminating co-occurrence of general and specific categories such as ”animal” and ”dog”). Finally, we randomly sample 1,000 classes from the remaining pool to construct ImageNet-CIL-1K. 50 images are randomly selected from every class in ImageNet-CIL-1K to form the validation set, while all remaining images are allocated to the training set. All images are resized to 224x224 pixels to reduce storage space. The resulting benchmark dataset comprises 1,000 classes, with 1,069,563 training images and 50,000 validation images, enabling comprehensive evaluations of continual learning methods on long task sequences.

Task Configuration. To allow for performance comparison across a range of task sequence lengths, we evaluate on sequences of 50, 75, and 100 tasks, where each task contains 10 classes. In accordance with (Rebuffi et al., 2017) and standard protocols, classes are first arranged in a randomized order (seed 1993) and subsequently partitioned into tasks for class-incremental learning. We employ exemplar-free learning, which does not allow access to data from previously learned tasks. The performance metric is the overall test accuracy over all classes encountered once a complete sequence of tasks has been learned.

3.2 Standard Benchmark Datasets

To demonstrate the generalizability of our method beyond long task sequence scenarios, we evaluate on standard continual learning benchmark datasets following established protocols. We conduct 10-task class-incremental learning experiments on four widely used datasets: CIFAR-100 (Krizhevsky et al., 2009), ImageNet-R (Hendrycks et al., 2021a), ImageNet-A (Hendrycks et al., 2021b), and OmniBenchmark (Zhang et al., 2022). In these experiments, we employ the original ImageNet-21K pretrained ViT-B/16 (Dosovitskiy et al., 2020) as the backbone and perform full model fine-tuning. This configuration is meant to validate that our sparsity-driven learning framework maintains competitive performance in conventional settings beyond long task sequence scenarios. Following (Rebuffi et al., 2017), we shuffle the class order using a random seed of 1993 for all methods.

3.3 Benchmark Architectures

We adopt two different architectural configurations corresponding to two experimental settings.

Config. #1. For standard continual learning benchmark datasets (e.g., 10 tasks), we use a standard ViT architecture following previous work (Zhou et al., 2024).

Config. #2. For long task sequences (e.g., 100 tasks), we propose an enhanced variant of ViT: starting from a pretrained backbone with U layers in total, we simply increase the number of channels in the final L layers. Our technical motivation is straightforward. It is widely observed that early layers in deep networks capture low-level features (e.g., edges and textures), which are general and transferable across tasks. In contrast, deeper layers encode high-level semantic information that tends to be more task-specific. In continual learning with long task sequences, accommodating diverse high-level representations becomes critical. Hence, expanding deeper layers enhances the model's capacity for channel mixing and high-level semantic understanding, thus mitigating parameter interference during continual learning. Note that while more advanced architectural modifications exist, designing a powerful neural architecture is not the main focus of our work. Instead, we adopt a simple yet effective modification solely to establish a minimal and reproducible testbed for studying long-sequence continual learning.

Note that the above architectural enhancement resonates with existing understandings of nervous system development. The human brain develops low-level sensory processing circuits (e.g., in the visual cortex) rapidly during early critical periods, after which these circuits become relatively stabilized (Stiles & Jernigan, 2010). In contrast, higher-order association areas in the brain remain plastic for a longer period, exhibiting ongoing connectivity reorganization and increasing processing capacity (Stiles & Jernigan, 2010), paralleling our augmented final L layers.

4 Experiments

We conduct comprehensive experiments to validate our sparsity-driven continual learning framework across two settings: (1) long task sequence scenarios with up to 100 tasks using our new ImageNet-CIL-1K benchmark, and (2) standard 10-task evaluations on established datasets. We further provide ablation studies analyzing each component's contribution to overall performance.

4.1 Baseline Methods

Continual Learning Baselines. For our long task sequence evaluation, we compare against recent state-of-the-art methods spanning different paradigms: prompt-based approaches (L2P (Wang et al., 2022b)), prototype-based methods (APER (Zhou et al., 2025)), adapter-based methods (EASE (Zhou et al., 2024), MoAL (Gao et al., 2025), SEMA (Wang et al., 2025)), model merging methods (MagMax (Marczak et al., 2024), Random Masking (Ke et al., 2024; Yu et al., 2024b), MoAL (Gao et al., 2025)), and a state-of-the-art subnetwork method (PGM (Wan & Yang, 2025)). We also include classical continual learning methods (LwF (Li & Hoiem, 2017), iCaRL (Rebuffi et al., 2017)) to illustrate the challenges of long sequences for traditional approaches.

Reference Baselines. To establish performance bounds, we include several reference methods: sequential fine-tuning (lower bound), linear probing (feature quality assessment), joint fine-tuning (upper bound), and CLIP zero-shot (Radford et al., 2021) (task-agnostic baseline). All trainable baselines utilize our enhanced backbone architecture to ensure fair comparisons.

4.2 Implementation Details

We evaluate our framework on multiple benchmarks using two backbone configurations: an ImageNet-21K pretrained ViT-B/16 (Config #1 in Section 3.3) for standard benchmarks (CIFAR-100, ImageNet-R, etc.), following the protocol of prior work (Zhou et al., 2024; Gao et al., 2025), and an enhanced ViT-B/16 backbone for our novel ImageNet-CIL-1K benchmark (Config #2 in Section 3.3), where the final layers are expanded and pretrained on ImageNet-1K. This enhanced configuration is also benchmarked from CLIP (Radford et al., 2021) initialization. Key hyperparameters, including the sparsity ratio $\rho \in [4, 15]$ and saturation point F, were set based on the estimated parameter requirements per task and their reuse frequency; comprehensive implementation details are deferred to the Appendix A.1.

Table 1: Comparison of different methods on ImageNet-CIL-1K. The columns under "# Tasks" represent the average test set accuracy of all learned classes after learning 100, 75, and 50 tasks, respectively, with each task having 10 classes. All models have been adjusted to have comparable number of trainable parameters.

Method	# Trainable Params (M)	# Tasks 100	75	50
Joint Finetune	195	73.7	-	-
Linear Probing	1	52.1	55.0	58.2
iCaRL (CVPR 2017)	195	36.7	42.8	46.1
LwF (TPAMI 2018)	195	27.6	29.7	33.6
Sequential Finetune	195	8.7	9.3	9.8
Random Masking (Arxiv 2024)	195	28.1	34.7	44.3
L2P (CVPR 2022)	-	39.7	47.3	52.1
EASE (CVPR 2024)	194	44.8	52.3	57.8
Subnetwork-PGM (ICML 2025)	193	36.5	40.6	44.8
MoAL (CVPR 2025)	196	52.9	56.7	62.0
APER (IJCV 2025)	196	55.3	58.1	60.3
MagMax (ECCV 2024)	194	53.1	56.9	59.2
Ours	195	**59.4**	**62.0**	**65.9**

4.3 Long Task Sequence CIL Results

Performance Comparison and Analysis. As shown in Table 1, our method consistently outperforms all baselines across task sequence lengths ranging from 50 to 100. On 50 tasks, our method achieves a significant improvement of 3.9% over the strongest baseline. This performance advantage is maintained as the task sequence scales, with improvements of 5.3% and 5.1% over MoAL and MagMax on 75 tasks, respectively. Finally, on the challenging 100-task sequence, our method surpasses MagMax, APER, and MoAL by 6.3%, 4.1%, and 6.5% in accuracy, respectively. These results show that our method achieves optimal plasticity-stability trade-off by maintaining superior performance across various task lengths, setting a new baseline for long-sequence CIL.

Evaluation with CLIP-pretrained Backbone. To assess the generalizability of our approach across different pretraining paradigms, we compare it against CLIP using the enhanced CLIP-pretrained ViT-B/16 backbone (Config #2, Section 3.3). As summarized in Table 2, our method exhibits consistent improvements over both continual learning and CLIP zero-shot baselines.

Table 2: Performance comparison using CLIP pretrained backbone model. The rightmost columns show performance on long task sequences with varying numbers of incremental tasks.

Method	CIFAR100	ImageNet-R	ImageNet-CIL-1K (# Tasks) 100	75	50
CLIP Zero-shot ViT-B/16 (86M)	68.7	77.1	61.7	60.7	61.9
CLIP Zero-shot ViT-L/14 (307M)	72.9	79.7	64.7	67.4	69.0
APER (IJCV 2025)	86.5	74.6	59.1	63.9	67.3
MoAL (CVPR 2025)	90.7	79.8	57.5	62.2	69.7
SEMA (CVPR 2025)	90.1	78.3	56.1	61.5	68.2
Ours	**91.0**	**80.3**	**64.9**	**69.1**	**73.9**

Notably, while APER, MoAL, and SEMA perform competitively or better than CLIP zero-shot ViT-B/16 (86M) on CIFAR100 (10 tasks), ImageNet-R (10 tasks), and medium sequences (50–75 tasks, ImageNet-CIL-1K), they fall short on the 100-task benchmark. In contrast, despite using a backbone with only 256M parameters, our method surpasses CLIP zero-shot ViT-L/14 (307M) in all task lengths, with a significant improvement of 4.9% and 1.7% on 50 and 75 tasks, respectively. These findings underscore the capacity of our method to harness powerful pretrained models.

4.4 Standard Benchmark Evaluation

We evaluate our method on established 10-task class-incremental benchmarks (CIFAR-100, ImageNet-R, ImageNet-A, OmniBenchmark) using a vanilla ViT-B/16 pretrained on ImageNet-21K to demonstrate its competitiveness in conventional settings. Table 3 shows that our method achieves state-of-the-art or competitive performance across all datasets: CIFAR-100 (90.7%, best performance), ImageNet-R (79.6%, best performance, +0.3% over MoAL), OmniBenchmark (78.7%, best performance), and ImageNet-A (64.0%, comparable to MoAL's 64.1%).

Table 3: Comparison of test accuracy on standard benchmark datasets each split into 10 tasks using ImageNet-21k pretrained ViT-B/16 backbone.

Method	CIFAR100	ImageNet-R	ImageNet-A	OmniBenchmark
Sequential Finetune	82.1	68.6	40.6	62.4
LwF (TPAMI 2018)	77.6	69.6	40.2	64.6
L2P (CVPR 2022)	84.5	73.7	45.5	63.8
EASE (CVPR 2024)	87.3	69.2	76.0	74.4
APER (IJCV 2025)	85.8	72.1	55.7	73.3
MoAL (CVPR 2025)	90.5	79.3	**64.1**	78.6
Ours	**90.7**	**79.6**	64.0	**78.7**

Importantly, our method demonstrates superior performance across both short and long task sequences: it significantly outperforms MoAL (the strongest baseline on short sequences) by 6.5% on 100 tasks (Table 1) and also substantially surpasses APER (the strongest baseline on long sequences) by 8.3% on 10-task ImageNet-A (Table 3). This dual competence across both standard and long task sequences validates the general applicability of our sparsity-driven framework, establishing it as a unified solution for diverse CIL scenarios.

4.5 Ablation Studies

We conduct ablation studies to validate all components of our sparsity-driven learning framework. All experiments are performed on 100-task sequences using our ImageNet-CIL-1K dataset and expanded ViT-B/16. Ablations on the impact of random forgetting and architecture setting are given in the Appendix A.2.

Table 4: Ablation study on sparsity constraints and training strategies. All variants use 195M trainable parameters and are evaluated after 100 tasks. The full framework uses a learned sparse mask with a two-stage training process.

Variant	Accuracy (%)	Difference (%)
Full Framework (Ours)	**59.4**	-
Random Mask for Stage 2	47.5	-11.9
No Regularization (Stage 1 Only)	36.9	-22.5
L1 Regularization (Stage 1 Only)	41.3	-18.1
No Regularization (2 Stage)	51.7	-7.7
L2 Regularization (2 Stage)	34.7	-24.7
w/o Gaussian Noise	58.0	-1.4
Gaussian Noise $\sigma = 0.005$	56.8	-2.6
Gaussian Noise $\sigma = 0.0001$	59.1	-0.3
Gaussian Noise $\sigma = 0.0005$ (Default)	59.4	-

Sparsity Mechanism Analysis. Table 4 show the critical role of both learned sparsity and the two-stage mask-task learning separation. Replacing our learned masks with random masks of equivalent sparsity causes an 11.9% performance drop (59.4% to 47.5%), confirming the importance of strategic parameter selection. The necessity of sparsity during task learning is evident when removing it entirely (No Regularization stage 1 only): accuracy plummets 22.5% to 36.9%, showing severe parametric interference without sparsity constraint.

One-stage fine-tuning based on both L_1 regularization and cross-entropy loss proves even less effective (41.3%, -18.1%), validating our decoupled mask discovery and task learning strategy. L2 regularization instead of L1 performs similarly poorly (34.7%, -24.7%), indicating that weight magnitude control cannot substitute for explicit sparsity enforcement.

Gaussian noise injection provides moderate but consistent improvements, with an optimal standard deviation of $\sigma = 0.0005$. Removing noise causes 1.4% performance loss, confirming its role in interference mitigation.

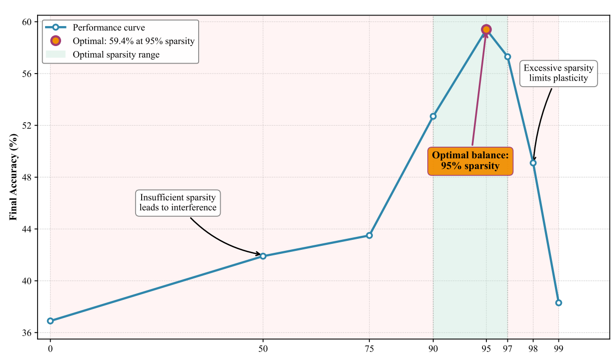

Figure 3: Impact of sparsity ratio on test accuracy after 100 tasks. Optimal accuracy (59.4%) occurs at 95% sparsity, with sub-optimal results at both lower and higher sparsity levels. The green zone denote the optimal range balancing interference reduction and learning plasticity.

Figure 3 reveals that 95% sparsity optimally balances interference reduction with learning plasticity, with performance degrading at both extremes. Insufficient sparsity leads to severe interference, whereas excessive sparsity leads to limited learning plasticity.

5 RELATED WORK

Continual Learning Traditional continual learning methods address catastrophic forgetting through regularization constraints (Kirkpatrick et al., 2017; Li & Hoiem, 2017), rehearsal strategies that retain exemplars of previous tasks (Rebuffi et al., 2017), or sparse dynamic parameter allocation techniques inspired from model pruning and sparse learning (Mallya & Lazebnik, 2018; Wang et al., 2022a; Yildirim et al., 2024; Wan & Yang, 2025). The advent of large-scale pretrained models has catalyzed a paradigm shift toward *pretrained model (PTM)-based continual learning*, which leverages backbone representations and employs adapters (Zhou et al., 2024; Yu et al., 2024a; Wang et al., 2025; Zhou et al., 2025; Gao et al., 2025), prompts (Wang et al., 2022b; Smith et al., 2023), or model merging strategies (Marczak et al., 2024; Ke et al., 2024).

Adapter-based methods can be categorized into router-based methods, which maintain task-specific adapters and employ routing mechanisms for adapter selection during inference, and prototype-based methods, which project prototype features from previous tasks into new feature spaces. However, both categories exhibit a scalability bottleneck: router-based methods suffer from increasing routing errors as task sequence lengths grow, while prototype-based methods accumulate projection errors across extended sequences, leading to performance degradation. In contrast, model merging methods bypass these architectural constraints by directly integrating task-specific knowledge into a unified model without requiring complex routing or projection mechanisms. Our work extends PTM-based continual learning to long task sequence scenarios, a critical yet underexplored domain.

Model Merging Model merging methods seek to combine multiple specialized models, often by transferring parameter updates from task-specific models to a shared base model (Yu et al., 2024b; Marczak et al., 2024; Ke et al., 2024; Gao et al., 2025). To mitigate interference among tasks, recent methods apply post-hoc sparsity to these parameter updates. DARE (Yu et al., 2024b) and its concurrent application to continual learning (Ke et al., 2024) randomly prune a large fraction of updates. In contrast, MagMax (Marczak et al., 2024) only retains the update with the largest magnitude for each parameter, mitigating interference but leading to insufficient plasticity. Meanwhile, MoAL (Gao et al., 2025) does not consider parameter interference and merges multiple task-specific adapters into a shared one using an experiential moving average of their parameters.

The common thread in these works is the application of a predefined sparsity pattern (random or magnitude-based) after task-specific training. Our work departs from this approach by explicitly learning sparse masks as an integral part of the training process. This allows our framework to proactively discover a sparse set of most relevant parameters for each task prior to training and prepare each task-specific model against interference during training.

6 LIMITATIONS

To the best of our knowledge, this is the first piece of work that scales continual learning up to 100 non-trivial vision tasks. While we recognize that exploring even longer task sequences is of significant interest for real-world applications, such extensive evaluation reaches beyond the scope of this study due to computing resource limitations. However, given the simplicity and promising performance of our method on 100 tasks, we believe that our method can serve as a capable baseline in continual learning under long-sequence settings. We hope that further work can explore the potential of our method for longer task sequences. Possible future improvements include further mitigating parameter interference to reduce catastrophic forgetting and scaling the number of trainable parameters proportionally with the number of tasks to dynamically enhance model capacity.

7 CONCLUSION

In this paper, we have tackled the formidable problem of long-sequence continual learning by addressing the core issue of balancing catastrophic forgetting and learning plasticity via a sparsity-driven learning framework. Our approach, validated on a new challenging benchmark dataset of 100 tasks, significantly advances the state of the art. We believe this work is a significant step towards scalable continual learning systems.

References

Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 139–154, 2018.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

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.

David L Donoho and Benjamin F Logan. Signal recovery and the large sieve. *SIAM Journal on Applied Mathematics*, 52(2):577–591, 1992.

David L Donoho and Philip B Stark. Uncertainty principles and signal recovery. *SIAM Journal on Applied Mathematics*, 49(3):906–931, 1989.

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.

Hermann Ebbinghaus. *Memory: A Contribution to Experimental Psychology*. Dover Publications, 1964.

Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal gradient descent for continual learning. In *International conference on artificial intelligence and statistics*, pp. 3762–3773. PMLR, 2020.

Zijian Gao, Wangwang Jia, Xingxing Zhang, Dulan Zhou, Kele Xu, Feng Dawei, Yong Dou, Xinjun Mao, and Huaimin Wang. Knowledge memorization and rumination for pre-trained model-based class-incremental learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 20523–20533, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8340–8349, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15262–15271, 2021b.

Hai-Jian Ke, Kun-Peng Ning, Yu-Yang Liu, Jia-Yu Yao, Yong-Hong Tian, and Li Yuan. Sparse orthogonal parameters tuning for continual learning. *arXiv preprint arXiv:2411.02813*, 2024.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

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

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017.

Christos Louizos, Max Welling, and Diederik P Kingma. Learning sparse neural networks through l_0 regularization. *International Conference on Learning Representations*, 2018.

Rongrong Ma, Jianyu Miao, Lingfeng Niu, and Peng Zhang. Transformed l_1 regularization for learning sparse deep neural networks. *Neural Networks*, 119:286–298, 2019. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2019.08.015. URL https://www.sciencedirect.com/science/article/pii/S0893608019302321.

Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 7765–7773, 2018.

Daniel Marczak, Bartłomiej Twardowski, Tomasz Trzciński, and Sebastian Cygert. Magmax: Leveraging model merging for seamless continual learning. In *European Conference on Computer Vision*, pp. 379–395. Springer, 2024.

George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11): 39–41, 1995.

Huafeng Qin, Xin Jin, Hongyu Zhu, Hongchao Liao, Mounîm A El-Yacoubi, and Xinbo Gao. Sumix: Mixup with semantic and uncertain information. In *European Conference on Computer Vision*, pp. 70–88. Springer, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 2001–2010, 2017.

Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. Imagenet-21k pretraining for the masses. *arXiv preprint arXiv:2104.10972*, 2021.

James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11909–11919, 2023.

Joan Stiles and Terry L Jernigan. The basics of brain development. *Neuropsychology review*, 20(4): 327–348, 2010.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pp. 10347–10357. PMLR, 2021.

Fengqiang Wan and Yang Yang. Probabilistic group mask guided discrete optimization for incremental learning. In *Forty-second International Conference on Machine Learning*, 2025.

Huiyi Wang, Haodong Lu, Lina Yao, and Dong Gong. Self-expansion of pre-trained models with mixture of adapters for continual learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 10087–10098, 2025.

Zifeng Wang, Zheng Zhan, Yifan Gong, Geng Yuan, Wei Niu, Tong Jian, Bin Ren, Stratis Ioannidis, Yanzhi Wang, and Jennifer Dy. Sparcl: Sparse continual learning on the edge. *Advances in Neural Information Processing Systems*, 35:20366–20380, 2022a.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 139–149, 2022b.

Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. *Advances in neural information processing systems*, 29, 2016.

Murat Onur Yildirim, Elif Ceren Gok, Ghada Sokar, Decebal Constantin Mocanu, and Joaquin Vanschoren. Continual learning with dynamic sparse training: Exploring algorithms for effective model updates. In *Conference on parsimony and learning*, pp. 94–107. PMLR, 2024.

Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Ping Hu, Dong Wang, Huchuan Lu, and You He. Boosting continual learning of vision-language models via mixture-of-experts adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23219–23230, 2024a.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*, 2024b.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pp. 3987–3995. PMLR, 2017.

Yuanhan Zhang, Zhenfei Yin, Jing Shao, and Ziwei Liu. Benchmarking omni-vision representation through the lens of visual realms. In *European conference on computer vision*, pp. 594–611. Springer, 2022.

Da-Wei Zhou, Hai-Long Sun, Han-Jia Ye, and De-Chuan Zhan. Expandable subspace ensemble for pre-trained model-based class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23554–23564, 2024.

Da-Wei Zhou, Zi-Wen Cai, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. *International Journal of Computer Vision*, 133(3):1012–1032, 2025.

A Appendix

A.1 Implementation Details

Training Details. All experiments use PyTorch on NVIDIA H800 GPUs. For experiments on our ImageNet-CIL-1K benchmark dataset (Config #2 in Section 3.3), we expand the final $L = 3$ layers in ViT-B/16 from 768 to 2304 dimensions with SiLU gating. Our enhanced ViT-B/16 backbone is pretrained on ImageNet-1K following standard hyperparameters set in (Liu et al., 2021) before being fine-tuned on continual learning tasks. While standard continual learning practices typically employ ImageNet-21K pretrained models (Zhou et al., 2024; 2025; Gao et al., 2025), we deliberately choose ImageNet-1K pretrained ViT-B/16 (Touvron et al., 2021) to ensure completely disjoint data distributions between pretraining and our benchmark dataset derived from ImageNet-21K. This prevents potential data leakage, providing a more rigorous evaluation of continual learning capabilities. Our two-stage fine-tuning employs Stage 1 mask discovery with L1 regularization ($\lambda = 8 \times 10^{-5}$, learning rate 2×10^{-4}) followed by Stage 2 mask-constrained fine-tuning with cross-entropy loss (80 epochs, learning rate 3^{-4}). We enforce 95% sparsity ($\rho = 95\%$) through percentile thresholding, and apply gradual random forgetting after $F = 60$ tasks with forgetting rate $1/F = 1.67\%$ to the accumulated parameter updates. Gaussian noise ($\sigma = 0.0005$) is injected into the frozen parameters during Stage 2 forward passes. Training uses AdamW optimizer with cosine scheduling, weight decay set to 10^{-2}, total batch size set to 512, and standard ImageNet training augmentations (Liu et al., 2021). The sparsity ratio ρ and the saturation point F are set according to the estimated average number of parameters required for each task and the estimated number of times a parameter can be reused, respectively.

We also benchmark our approach on ImageNet-CIL-1K with CLIP (Radford et al., 2021) (Config #2 in Section 3.3). For a fair comparison, we initialize our model using a CLIP-pretrained ViT-B/16 backbone and apply the architectural enhancement introduced in Section 3.3. Since the newly appended parameters have not been pretrained, we freeze all pretrained layers and only train the expanded layers on ImageNet-1K to obtain a stable starting point. This fine-tuning step uses the standard ImageNet-1K training setting (Liu et al., 2021). For continual learning, we mostly use the same hyperparameters in the above ImageNet-1K pretrained setting and made the following changes to achieve stronger performance: $\lambda = 2 \times 10^{-4}$, 200 training epochs with a learning rate of 9×10^{-5}, $\sigma = 0.0003$, $F = 100$, and leveraged SuMix (Qin et al., 2024) as an additional data augmentation.

For a fair evaluation on the standard continual learning benchmark datasets (CIFAR100, ImageNet-R, ImageNet-A, OmniBenchmark) using Config #1 in Section 3.3, we adopt an ImageNet-21K pretrained ViT-B/16 backbone following (Zhou et al., 2024; Gao et al., 2025). The training hyperparameters and setting follow (Gao et al., 2025) and the hyperparameters for our sparsity-driven learning framework are $\lambda = 7 \times 10^{-5}$, $\rho = 85\%$, $\sigma = 0.0006$, and $F = 10$ (no forgetting).

Ensemble Output. Our merged backbone model is integrated with a multi-component ensemble for final prediction, combining three existing techniques to address inference-time task identification in long-sequence continual learning. Specifically, we employ: (1) a mixture-of-experts (MoE) approach from (Yu et al., 2024a) that maintains T separate classification heads corresponding to the learned tasks, where each head h_t is trained using our sparse parameter updates while others remain frozen to ensure task-specific specialization, (2) autoencoder-based task identification following (Yu et al., 2024a) that computes reconstruction scores from task-specific autoencoders to identify the most likely source task during inference, with final task selection based on the average between autoencoder reconstruction scores and top-2 MoE predictions for robust routing, and (3) a separate classification head from MoAL (Gao et al., 2025) that maintains class prototypes. The final prediction averages outputs from the selected MoE classification head and prototype-based prediction. Ablation studies in Section 4.5 illustrate the individual contribution of each ensemble component. Note that this ensemble is not part of our main contributions in this paper, but an integral part of our overall continual learning pipeline.

A.2 Additional Ablation Studies

Table 5: Comparison between independent and sequential fine-tuning in our sparsity-driven learning framework.

Method	Accuracy
Independent fine-tune (Ours)	59.4
Sequential fine-tune with sparse mask	50.9

Impact of Independent Fine-tuning. In our sparse learning framework, we fine-tune the base model for each task independently using the two stage framework introduced in Section 2.1. In Table 5, we observe that sequential fine-tuning, where the base model is the accumulated θ_{accu} in Equation 2 instead of θ_{base}, lead to sub-optimal results. This can be attributed to the fact that the model fine-tuned on the last task is equivalent to the final merged model, leading to biased performance towards tasks learned later on and substantial interference with previous learned tasks, resulting in catastrophic forgetting.

Random Forgetting Impact. Fig 4 illustrate the impact of the selection of saturation point F (also can be thought of as the forgetting rate) in our forgetting mechanism. The optimal 1.67% forgetting rate prevents interference accumulation while preserving essential knowledge. No forgetting ($F = 100$, 0% forgetting rate) causes performance degradation due to unchecked parameter conflicts, while excessive forgetting (2.0% or more) leads to knowledge loss, confirming the principled calculation of our forgetting rate.

Parameter Efficiency Analysis. Our enhanced ViT-B/16 backbone increases the number of trainable parameters to 195M. Here, we investigate the performance of our approach and representative continual learning baselines when modifying the number of trainable parameters. Table 6 reveals superior parameter utilization compared to existing methods. Scaling from 50M to 195M trainable

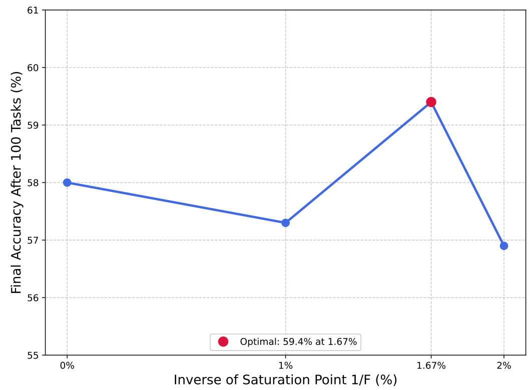

Figure 4: Effect of random forgetting ratio on test accuracy after 100 tasks. No forgetting (0%) causes parameter interference, while excessive forgetting (2%) leads to knowledge loss.

Table 6: Performance comparison with a varying number of trainable parameters.

	# Trainable Params (M)		
Method	50	100	195
Random masking	26.3	27.9	28.1
MoAL	51.6	52.1	52.9
MagMax	50.9	52.3	53.1
Ours	54.0	56.7	59.4

parameters by changing the number of channels in either our backbone or the adapter, our method improves 5.4% while competing methods show minimal gains: Random Masking (+1.8%), MoAL (+1.3%), and MagMax (+2.2%). This demonstrates that our sparsity-driven framework effectively leverages additional parameter capacity while maintaining interference control.

Architectural Component Contributions. Our enhanced ViT-B/16 has 195M trainable parameters in the last L (empirically set to 3) layers of the network by setting the channel size to 2304. Additionally, we introduce a gating mechanism into the augmented final L layers to enable dynamic selection of relevant information during continual learning. Mathematically, given an input X, the gating mechanism is defined as $\text{SiLU}\,(WX) \odot \text{SiLU}\,(X)$, where W is the weight matrix.

Table 7 validates our architectural modifications for long-sequence continual learning through fair comparison that maintains the same number of trainable parameters (195M) for all variants. The choice of fine-tuning depth is important: two layers provide insufficient plasticity (-7.6% to 51.8%), while four layers introduce excessive interference (-4.2% to 55.2%). Our three-layer configuration achieves the optimal plasticity-stability balance. Full model fine-tuning without freezing any layers substantially degrades performance (-5.3% to 54.1%), confirming that selective layer adaptation prevents interference with general low-level representations. Removing the SiLU gating mechanism reduces performance by 1.8% (57.6%), demonstrating its importance for dynamic parameter weighting.

Table 7: Ablation study on architectural enhancements. All variants use 195M trainable parameters.

Method Variant	Accuracy
Ours	59.4
Last 2 Layers Fine-tuned	51.8
Last 4 Layers Fine-tuned	55.2
Full Model Fine-tuning	54.1
w/o Gating Mechanism	57.6

Ensemble Component Analysis. Table 8 reveals the synergistic effects of our output ensemble combining MoE, autoencoder, and MoAL components. Individual components show substantial performance gaps when used in isolation: 57.8% (MoAL alone), 56.1% (MoE alone), and 57.0% (autoencoder alone). However, we find that even when these individual components are used in isolation, our method still outperforms other CIL methods. The complete ensemble achieves 59.4%, confirming that the complementary strengths of individual components contribute to optimal long task sequence performance through diverse representation and prediction strategies.

Table 8: Ablation study on classifier components. Removing any component reduces performance, and individual components alone are substantially weaker than their integration.

Method Variant	Accuracy
Full Framework (Ours)	59.4
w/o Autoencoder	58.5
w/o MOE	58.7
w/o MoAL	58.3
MoAL Alone	57.8
MOE Alone	56.1
Autoencoder Alone	57.0

Table 9: Comparison of training speed between model merging methods

Method	Training time
MoAL	16.25 minutes
MagMax	12.17 minutes
Random Masking	12.03 minutes
Ours	15.61 minutes

A.3 Comparison of Training Speed

Since our method takes the model merging approach, we compare the training time with model merging baselines. Despite our two-stage sparse learning framework, our optimized implementation has a reasonable computational cost compared to recent model merging methods when training on the ImageNet-CIL-1K benchmark dataset. In particular, Stage 1 is only required to learn the sparse mask so we can reduce the number of training epochs. Since Stage 2 only optimizes a sparse subset of parameters, its training time is approximately 30% that of Stage 1. Therefore, the overall training complexity is comparable to existing model merging methods. Note that Random Masking and MagMax have similar steps in the training procedure so their training times are approximately the same.

A.4 Sparsity-driven continual learning algorithm

The pseudo code for our two-stage sparsity-driven learning framework is given in Algorithm 1.

A.5 New Benchmark Dataset ImageNet-CIL-1K

This section details the composition and key statistics of the proposed ImageNet-CIL-1K benchmark. The dataset comprises 1,000 classes, with a near-uniform distribution of images per class. As illustrated in Figure 5, the number of images per class has a mean of 1069.6 and a median of 1098.5, indicating a balanced and robust dataset suitable for large-scale continual learning evaluation.

A complete listing of all 1,000 class names is provided for reference.

Algorithm 1 Sparsity-Driven Continual Learning Algorithm

Require: pretrained base model θ_{base}, task sequence $\{\mathcal{S}_k\}_{k=1}^{T}$, sparsity ratio ρ, noise variance σ^2, saturation point F
Ensure: Final merged model θ_{merged}

Initialize $\Delta\theta_{\text{acc}} \leftarrow \mathbf{0}$
for each task $k \in \{1, \ldots, T\}$ **do**
  Initialize $\Delta\theta_k \leftarrow \mathbf{0}$ ▷ Stage 1: Mask Learning
  Minimize $\mathcal{L}_{\text{mask}} = \mathbb{E}_{(x,y)\sim\mathcal{S}_k}[\ell(f(x;\theta_{\text{base}} + \Delta\theta_k), y)] + \lambda\|\Delta\theta_k\|_1$ via gradient descent
  $M_k^{(i)} \leftarrow \begin{cases} 1 & \text{if } |\Delta\theta_k^{(i)}| \geq \rho \text{ percentile of all } \Delta\theta_k \\ 0 & \text{otherwise} \end{cases}$
  Set `require_grad` $= (M_k^{(i)} == 1)$ for all i
  Initialize $\Delta\theta_k \leftarrow \mathbf{0}$ ▷ Stage 2: Fine-tuning
  while not converged **do**
    Sample noise $\epsilon^{(i)} \sim \mathcal{N}(0, \sigma^2)$ for positions where $M_k^{(i)} = 0$
    Set $\epsilon^{(i)} = 0$ for positions where $M_k^{(i)} = 1$
    Gradient descent step for $\min_{\theta_k} \mathbb{E}_{(x,y)\sim\mathcal{S}_k}[\ell(f(x;\theta_{\text{base}} + \epsilon + \Delta\theta_k), y)]$
    ▷ Cross-entropy only
  end while
  if $k > F$ **then** ▷ Gradual random forgetting
    Generate random binary mask $\mathcal{F}$ with forgetting rate $1/F$
    Apply forgetting: $\Delta\theta_{\text{acc}} \leftarrow \mathcal{F} \odot \Delta\theta_{\text{acc}}$
  end if
  $\Delta\theta_{\text{acc}} \leftarrow \Delta\theta_{\text{acc}} + \Delta\theta_k$ ▷ Add current task update
end for
$\theta_{\text{merged}} \leftarrow \theta_{\text{base}} + \Delta\theta_{\text{acc}}$ ▷ Final merge **return** θ_{merged}

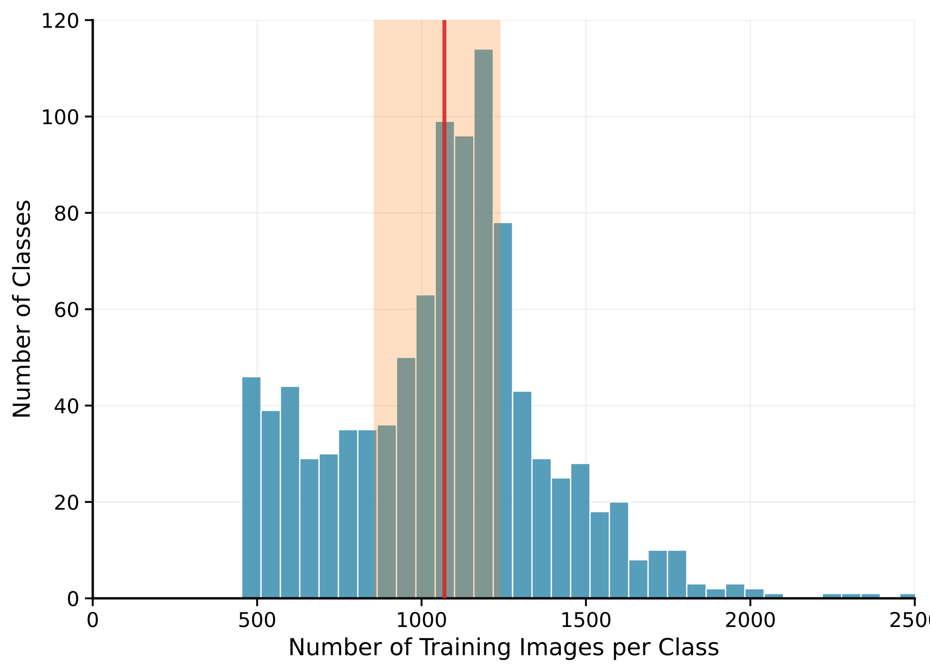

Figure 5: Histogram of image count per class in the proposed ImageNet-CIL-1K benchmark dataset. The distribution shows a high concentration around the mean (1069.6, denoted by the red line; orange region denote the inter-quartile range), confirming a balanced allocation of samples across classes.

A.6 Ethics Statement

This work presents an advancement in continual learning algorithms. We use standard, publicly available datasets for training and evaluation. Our research does not raise any ethical issues, as it is not directed towards any specific application domain that could be deemed harmful (e.g., surveillance, misinformation).

A.7 Reproducibility Statement

Although the code is not provided together with this paper submission, we are committed to making the source code publicly available once the paper receives a final decision. Comprehensive implementation details and the pseudo code for our sparse learning framework are provided to aid reproducibility.

A.8 Use of LLM.

Deepseek-r1 (Guo et al., 2025) and Gemini 2.5 pro (Comanici et al., 2025) have been used solely to check for grammatical errors, polish the wording, and drawing figures with Python.

- piaffe
- rock climbing
- acrobatics
- broad jump
- bathe
- dip
- fight
- spar
- archery
- Greco-Roman wrestling
- rollerblading
- speed skating
- bullfighting
- ducking
- surf casting
- ice hockey
- professional golf
- shuffleboard
- professional football
- touch football
- rugby
- professional baseball
- no-hit game
- two-hitter
- lacrosse
- professional tennis
- doubles
- team sport
- feeder
- sire
- herpes simplex 1
- paramecium
- eukaryote
- striped killifish
- guppy
- soldierfish
- gastrula
- porbeagle
- great white shark
- sand tiger
- blacktip shark
- dusky shark
- smoothhound
- shovelhead
- dickeybird
- cassowary
- emu
- hedge sparrow
- meadow pipit
- brambling
- pine siskin
- house finch
- bullfinch
- dark-eyed junco
- white-crowned sparrow
- chipping sparrow
- evening grosbeak
- cardinal
- baya
- scrubbird
- phoebe
- cock of the rock
- ovenbird
- pitta
- spotted flycatcher
- ring ouzel
- wood thrush
- whinchat
- wheatear
- robin
- goldcrest
- blackcap
- greater whitethroat
- lesser whitethroat
- sedge warbler
- parula warbler
- Audubon's warbler
- myrtle warbler
- blackpoll
- raven
- blue jay
- Clark's nutcracker
- butcherbird
- house wren
- long-billed marsh wren
- red-breasted nuthatch
- white-breasted nuthatch
- blue tit
- tree swallow
- satin bowerbird
- Bohemian waxwing
- Cooper's hawk
- marsh harrier
- sparrow hawk
- bald eagle
- griffon vulture
- Egyptian vulture
- black vulture
- black vulture
- great grey owl
- tawny owl
- screech owl
- spotted owl
- Old World scops owl
- European fire salamander
- spotted salamander
- common newt
- red eft
- spotted salamander
- axolotl
- hellbender
- wood-frog
- green frog
- tailed frog
- American toad
- obstetrical toad
- canyon treefrog
- green turtle
- Atlantic ridley
- common snapping turtle
- diamondback terrapin
- red-bellied terrapin
- painted turtle
- tuatara
- banded gecko
- common iguana
- marine iguana
- side-blotched lizard
- tree lizard
- Texas horned lizard
- western skink
- agama
- frilled lizard
- mountain devil
- green lizard
- Komodo dragon
- African crocodile
- Chinese alligator
- gavial
- triceratops
- smooth green snake
- green snake
- corn snake
- gopher snake
- pine snake
- milk snake
- common garter snake
- ribbon snake
- water moccasin
- vine snake
- boa constrictor
- anaconda
- carpet snake
- copperhead
- hamadryad
- green mamba
- taipan
- sea snake
- horned viper

- Mojave rattlesnake
- massasauga
- ground rattler
- fer-de-lance
- harvestman
- orb-weaving spider
- black and gold garden spider
- garden spider
- cockerel
- brood hen
- pullet
- Orpington
- European black grouse
- capercaillie
- spruce grouse
- greater prairie chicken
- ring-necked pheasant
- hoatzin
- rock dove
- band-tailed pigeon
- Streptopelia turtur
- mourning dove
- roller
- popinjay
- poll
- kea
- sulphur-crested cockatoo
- budgerigar
- European cuckoo
- belted kingfisher
- Euopean hoopoe
- European swift
- frogmouth
- green woodpecker
- yellow-shafted flicker
- red-shafted flicker
- barbet
- toucanet
- quack-quack
- diving duck
- mallard
- black duck
- bufflehead
- mandarin duck
- eider
- American merganser
- red-breasted merganser
- Chinese goose
- honker
- coscoroba
- cob
- pen
- mute swan
- trumpeter
- tusker
- echidna
- opossum rat
- giant kangaroo
- rock wallaby
- tree wallaby
- numbat
- calf
- doe
- sea fan
- mushroom coral
- woodborer
- common limpet
- Hermissenda crassicornis
- tiger cowrie
- seashell
- ark shell
- chambered nautilus
- blue crab
- fiddler crab
- American lobster
- spiny lobster
- daphnia
- sacred ibis
- common spoonbill
- snowy egret
- black-crowned night heron
- yellow-crowned night heron
- crested cariama
- Florida gallinule
- European gallinule
- American gallinule
- Old World coot
- killdeer
- dotterel
- lapwing
- ruddy turnstone
- surfbird
- red-backed sandpiper
- redshank
- lesser yellowlegs
- curlew sandpiper
- sanderling
- upland sandpiper
- American woodcock
- great snipe
- European curlew
- black-necked stilt
- black-winged stilt
- pratincole
- black-backed gull
- laughing gull
- sea swallow
- skimmer
- great skua
- razorbill
- pigeon guillemot
- thick-billed murre
- Atlantic puffin
- white pelican
- solan
- water turkey
- Adelie
- king penguin
- rock hopper
- fulmar
- common dolphin
- killer whale
- manatee
- crabeater seal
- harp seal
- Chihuahua
- Maltese dog
- Shih-Tzu
- bluetick
- Italian greyhound
- Ibizan hound
- Norwegian elkhound
- Border terrier
- Irish terrier
- Norwich terrier
- wire-haired fox terrier
- Lakeland terrier
- Airedale
- Boston bull
- miniature schnauzer
- giant schnauzer
- golden retriever
- Labrador retriever
- vizsla
- English setter
- clumber
- Old English sheepdog
- Shetland sheepdog
- German shepherd

- miniature pinscher
- Greater Swiss Mountain dog
- Bernese mountain dog
- bull mastiff
- affenpinscher
- pug
- Pomeranian
- keeshond
- Cardigan
- standard poodle
- dingo
- kit fox
- kitty
- alley cat
- kitten
- Angora
- Egyptian cat
- cougar
- jungle cat
- bobcat
- jaguar
- cheetah
- sloth bear
- prey
- big game
- tiger beetle
- flea beetle
- Colorado potato beetle
- scarab
- green June beetle
- rhinoceros beetle
- rove beetle
- common louse
- wiggler
- yellow-fever mosquito
- Africanized bee
- black bee
- bumblebee
- cicada killer
- fire ant
- oriental cockroach
- giant water bug
- wheel bug
- stonefly
- doodlebug
- painted beauty
- red admiral
- viceroy
- purple emperor
- peacock
- sulphur butterfly
- tea tortrix
- tussock caterpillar
- fall armyworm
- death's-head moth
- luna moth
- forest tent caterpillar
- corn earworm
- cabbageworm
- edible sea urchin
- European rabbit
- European hare
- polar hare
- antelope squirrel
- eastern chipmunk
- chipmunk
- groundhog
- guinea pig
- chinchilla
- rock hyrax
- filly
- broodmare
- American saddle horse
- Arabian
- Lippizan
- bucking bronco
- buckskin
- cayuse
- plow horse
- Exmoor
- carthorse
- farm horse
- palomino
- burro
- common zebra
- Indian rhinoceros
- white rhinoceros
- collared peccary
- dogie
- yak
- Jersey
- gaur
- ewe
- ram
- lambkin
- Hampshire
- merino
- billy
- bezoar goat
- ibex
- gnu
- sassaby
- Thomson's gazelle
- bongo
- nyala
- bushbuck
- steenbok
- common eland
- gemsbok
- pronghorn
- Japanese deer
- fallow deer
- Arabian camel
- guanaco
- two-toed sloth
- ant bear
- pangolin
- world
- Homo sapiens sapiens
- silverback
- gibbon
- howler monkey
- African elephant
- coati
- giant panda
- game fish
- blue catfish
- pollack
- allice shad
- landlocked salmon
- chinook
- coho
- goosefish
- frogfish
- perch
- European perch
- northern pike
- pumpkinseed
- flame fish
- crevalle jack
- cardinal tetra
- porkfish
- red drum
- mulloway
- white croaker
- spotted weakfish
- parrotfish
- skipjack
- bonito
- blue marlin
- bowfin
- paddlefish
- gar
- stonefish
- queen triggerfish
- balloonfish
- abacus
- A battery
- Abbe condenser
- abbey
- accelerator
- acropolis
- adapter
- afterburner
- air conditioner
- aircraft engine
- air horn
- air terminal
- alehouse

- altar
- altazimuth
- alternator
- amphitheater
- amphora
- analyzer
- anastigmat
- anchor chain
- AND circuit
- anklet
- antiperspirant
- ao dai
- aperture
- aquaplane
- argyle
- armoire
- arterial road
- ashtray
- assembly
- assembly hall
- athletic sock
- atrium
- attache case
- audio CD
- autobahn
- autofocus
- automobile engine
- autoradiograph
- autostrada
- awning
- axle bar
- baby grand
- baby shoe
- back brace
- back porch
- backsword
- backup system
- balloon sail
- ballot box
- baluster
- bandbox
- bandoneon
- bangle
- banner
- baptismal font
- barbershop
- bareboat
- baritone
- basilica
- basinet
- bass clarinet
- bass drum
- bass guitar
- bassinet
- bath salts
- batik
- batting glove
- batting helmet
- beach towel
- beacon
- beanbag
- beaver
- Bedford cord
- bed jacket
- bedsitting room
- bedstead
- beer barrel
- beer glass
- beer hall
- belfry
- belt buckle
- bench press
- besom
- bib
- bicycle chain
- bicycle rack
- bicycle seat
- bier
- billboard
- binder
- binnacle
- biplane
- birdbath
- birdcage
- blackwash
- blender
- blue
- boat hook
- boathouse
- bobby pin
- bobsled
- bodice
- bookcase
- bookend
- boot
- bootlace
- Boston rocker
- bottle
- bowling alley
- bowling shoe
- bracer
- brake drum
- brake lining
- brake pad
- brake shoe
- brasserie
- bread-bin
- breakfast area
- breathalyzer
- broad arrow
- brochette
- buckram
- buckskins
- buffer
- bugle
- bullpen
- bulwark
- bungalow
- bunk bed
- bunker
- bunsen burner
- burqa
- bushel basket
- bustle
- butterfly valve
- cabinetwork
- cafeteria tray
- caftan
- caldron
- camisole
- campanile
- canopic jar
- canopy
- canteen
- canteen
- cantilever bridge
- cantle
- car
- caravansary
- car bomb
- cardroom
- carpenter's kit
- carpet sweeper
- carryall
- carrycot
- carving knife
- case knife
- cash machine
- Cassegrainian telescope
- cassette recorder
- catamaran
- cat box
- cathedral
- CD drive
- CD-R
- censer
- cereal box
- chafing dish
- chain
- chainlink fence
- chain store
- chair
- chalice
- chancel
- chancellery
- chateau
- chatelaine
- checkout
- cheekpiece
- chemise
- chemise
- chest protector
- chiffon
- chiffonier
- chin rest
- chukka
- churn
- cigar box
- cigarette butt
- circle
- circuit breaker
- city hall
- cleats
- cleaver
- clerestory
- climbing frame

- clinical thermometer
- clipper
- clock tower
- cloisonne
- cloister
- clothesbrush
- clothes tree
- coat button
- cockhorse
- cockleshell
- cockpit
- cocktail lounge
- cocktail shaker
- cocotte
- coffee can
- coif
- collar
- collet
- Colt
- columbarium
- combination lock
- command module
- compact-disk burner
- compound microscope
- compression bandage
- concert grand
- concert hall
- concrete mixer
- console table
- contact
- container ship
- control tower
- cooler
- corbel
- cords
- cork
- corner
- cornice
- country store
- courthouse
- covered bridge
- coverlet
- cowbell
- crampon
- crazy quilt
- cricket bat
- croquet mallet
- crucifix
- cupola
- curbstone
- dacha
- dairy
- dais
- dashiki
- data system
- davenport
- day school
- deck chair
- deep-freeze
- denim
- department store
- derrick
- desktop computer
- dessert spoon
- detached house
- dhow
- dialog box
- diaper
- digital clock
- digital subscriber line
- dining-room furniture
- DIP switch
- disk brake
- dispensary
- Dixie cup
- donkey jacket
- door
- doorplate
- dormer
- dovecote
- Dragunov
- drawer
- drawing room
- drawknife
- dredger
- dress hat
- dressing case
- dress suit
- driver
- dropper
- drum printer
- dry fly
- dry wall
- duckpin
- duffel
- dump truck
- dustcloth
- dustpan
- Eames chair
- earmuff
- easel
- eaves
- eggbeater
- eight ball
- elbow pad
- electric
- electrical cable
- electric locomotive
- electric typewriter
- electronic fetal monitor
- embassy
- encaustic
- English saddle
- ensign
- erecting prism
- espadrille
- etagere
- ethernet
- evening bag
- eyeliner
- face guard
- face powder
- fairy light
- false face
- fan blade
- fancy dress
- fan vaulting
- felucca
- fez
- field artillery
- field hockey ball
- field house
- fifth wheel
- figure skate
- finger
- finger paint
- fire bell
- fire screen
- firing chamber
- first class
- fixer-upper
- flagpole
- flintlock
- flip-flop
- float
- floatplane
- floor lamp
- florist
- floss
- flying buttress
- fob
- food court
- foredeck
- foremast
- foulard
- four-poster
- franking machine
- freewheel
- freight liner
- French horn
- Frisbee
- front projector
- fruit machine
- gabardine
- gaff topsail
- gag
- gaiter
- gambrel
- Garand rifle
- garter belt
- garter stitch
- gas gun
- gas holder
- gas oven
- gateleg table

- gharry
- ghat
- gift shop
- gift wrapping
- gig
- glebe house
- Global Positioning System
- gnomon
- goalpost
- golf bag
- golf glove
- golliwog
- gouge
- gown
- grab bag
- graduated cylinder
- grandfather clock
- grape arbor
- grater
- gravestone
- grey
- griddle
- grinder
- Guarnerius
- guided missile frigate
- gusset
- hair spray
- half binding
- hand glass
- hand lotion
- hand luggage
- hard hat
- harness
- harp
- hatpin
- hay bale
- headboard
- head gasket
- headpiece
- headset
- headstall
- hearse
- heat lamp
- hemostat
- hemstitch
- hideaway
- highchair
- hippodrome
- hockey stick
- home plate
- hone
- honeycomb
- horseshoe
- horseshoe
- hose
- hot tub
- hot-water bottle
- houseboat
- hula-hoop
- ice ax
- iced-tea spoon
- ice tongs
- igloo
- inclinometer
- incubator
- integrated circuit
- internal drive
- irons
- irrigation ditch
- jack-in-the-box
- jigsaw puzzle
- joystick
- jungle gym
- junk
- junk shop
- kachina
- kayak
- ketch
- khadi
- kilt
- kirtle
- kitchen table
- knee-high
- knitting machine
- knocker
- ladder-back
- ladder truck
- lag screw
- lame
- lancet window
- land line
- laser
- laser-guided bomb
- laser printer
- lawn chair
- lawn furniture
- leading rein
- leatherette
- lever lock
- lifeboat
- light pen
- Link trainer
- local
- lock
- log cabin
- long johns
- long sleeve
- loving cup
- LP
- luxury liner
- lyceum
- macrame
- magnum
- maillot
- mallet
- manhole
- manor
- manse
- marina
- masher
- mattress cover
- measuring cup
- meat grinder
- megaphone
- menhir
- microprocessor
- microtome
- microwave
- midiron
- miller
- minicar
- ministry
- minivan
- miter joint
- monkey-wrench
- monocle
- Moorish arch
- mortar
- mosaic
- motor scooter
- mountain bike
- mountain tent
- mouse
- mouthpiece
- mouthpiece
- mouthpiece
- movement
- mufti
- mule
- muzzle
- nailbrush
- nail polish
- narrow wale
- national monument
- neck brace
- needlenose pliers
- negative
- newspaper
- newsroom
- nipple
- nude
- nylons