A Simple Baseline that Questions the Use of Pretrained-Models in Continual Learning

Paul Janson[1,3] [*], Wenxuan Zhang[1], Rahaf Aljundi[2], and Mohamed Elhoseiny[1]

[1]King Abdullah University of Science and Technology, Saudi Arabia
[2]Toyota Motor Europe, Belgium
[3]University of Moratuwa, Sri Lanka

Abstract

With the success of pre-training techniques in representation learning, a number of continual learning methods based on pre-trained models have been proposed. Some of these methods design continual learning mechanisms on the pre-trained representations and only allow minimum updates or even no updates of the backbone models during the training of continual learning. In this paper, we question whether the complexity of these models is needed to achieve good performance by comparing them to a very simple baseline that we design. We argue that the pre-trained feature extractor itself can be strong enough to achieve a competitive or even better continual learning performance on Split-CIFAR100 and CoRe 50 benchmarks. To validate this, we conduct the baseline that 1) uses a frozen pre-trained model to extract image features for every class and computes their corresponding mean features during the training time, and 2) makes predictions based on the nearest neighbor distance between test samples and mean features of the classes; i.e., Nearest Mean Classifier (NMC). This baseline is single-headed, exemplar-free, and can be task-free by updating the mean features continually. This baseline achieved 88.53% on 10-Split-CIFAR-100, surpassing most state-of-the-art continual learning methods, with all initialized by the same pre-trained transformer model. We hope our baseline may encourage future progress in designing learning systems that can continually add quality to the learning representations even if they start from pre-trained weights. Code is available `https://github.com/Pauljanson002/pre-trained-cl.git`

1 Introduction

Conventional machine learning models struggle to perform well when the i.i.d. assumption is violated in the real world, where data arrives sequentially from tasks with shifting distributions over time. These models suffer from catastrophic forgetting of earlier tasks[15, 10]. Continual learning, in which the agent is expected to learn new tasks without forgetting the old ones, has been studied extensively as a potential solution to this issue. Early approaches start training with a model from scratch, continually adapt the model for future tasks, and prevent forgetting by replaying the data, designing the penalization of the updates of the model parameters, and/or dynamically increasing model parameters to incorporate new knowledge.

The use of pre-training [24] on large-scale datasets, such as ImageNet-1k[23] and ImageNet-21k [22] has led to significant advances. With sufficient and diverse data, the output features of a pre-trained model generalize well to a vast array of tasks, greatly increasing the performance of challenging learning scenarios such as continual learning. As a result, some existing works directly deploy the pre-trained models as feature extractors, and apply continual learning techniques on the feature

[*]Workdone during internship at KAUST

Workshop on Distribution Shifts, 36th Conference on Neural Information Processing Systems (NeurIPS 2022).

level. These methods imply that pre-trained models provide general and coarse features that need to be task-specifically fine-tuned. However, [20] stated large-scale pre-trained models have excellent classification performance to the data outside of the pre-trained distribution even without extra training. We are questioning whether these pre-trained features are competitive for downstream tasks and whether sophisticated continual learning techniques are indeed needed to achieve a good performance on the studied continual learning benchmarks.

To answer these questions, we implement a simple baseline in continual learning. We use a frozen pre-trained model to extract features for the training set and compute mean features to represent each class. During the testing time, we make predictions based on the distance between the test samples and class mean features. Our results are surprisingly outperforming SOTA methods using the same pre-trained backbone network. We achieve 88.53% Average Accuracy at the end of the learning sequence of Split CIFAR100 compared to 83.83% of L2P[27], and 83.23% on the evaluation task of CoRe50 compared to 78.33% of L2P. This implies that the pre-trained models provide quality and robust learning representations under distribution shifts. We also present the experimental results on two more diverse benchmarks, 5-dataset and Split-ImageNet-R. The results on these datasets also show competitive performance outperforming most methods.

Our proposed baseline is single-headed, task-free, and exemplar-free as an additional bonus. We argue that this is a simple yet powerful baseline that every continual learning method should compare against. The intentions of these properties are to make less use of task labels (which is unrealistic in practice) and replay data (which raises privacy concerns) in the techniques. We observe that some prior works outperform this baseline, but at the sacrifice of one or more of these characteristics. There are few comparable approaches with ours when all the conditions align. We hope that this work sheds some light on examining the practicality of pre-training in continual learning and whether the new methods are improving the learned representation quality continually.

2 Related Work

Continual learning with pre-trained models Continual learning methods generally trained feature extractors from scratch and constrained the drift in feature representation [10, 16]. Recently, the usage of pre-trained models has attracted more attention in continual learning. [2] viewed the first task as a pre-training stage and froze the feature representation after the first task. [7] found that a larger size of data in the first task and self-supervised pre-training helped to decrease catastrophic forgetting. [12] empirically analyzed the effect of training the last layer with a fixed feature extractor. [18] studied foundation models and replay of frozen latent features to overcome catastrophic forgetting. Recently [27] proposed to use prompt based fine-tuning with a pre-trained transformer [5, 25] and compared it with other methods in the same initialization. We adopt this strategy and point out that classifier learning actually reduced the power of representations learned by the pre-trained model.

Task-aware/free continual learning Task incremental method requires the task identifiers of the samples during the inference, which reduces the practical application of these methods, whereas class-incremental methods do not require those. However, task-aware/free methods suffer from class recency bias. To overcome such recency bias, BiC[29] proposed to add a final layer that reduces the final bias. LUCIR [9] proposed to use a balanced fine-tuning to train the classifier after freezing feature representation. Our method also focuses on this problem and uses the simple nearest mean classifier on top of pre-trained transformer.

Exemplar-free continual learning Earlier methods were proposed to store raw data, learned features, or generated features from previous tasks. Since the replay mechanism such as ER [4] is orthogonal to architecture-based and regularization-based methods, they are widely adopted in other methods to improve performance. Recently there has been an increased interest in exemplar-free continual learning to account for privacy concerns in storing raw samples and storage concerns. iCaRL [21] introduced the method of using exemplars for continual learning and selected exemplars close to the class means. Our baseline follows a similar strategy but only stores the class-mean in feature space which save the storage by reducing the number of latent variables to keep per class.

3 Methodology

Problem Setup We adopt the standard continual learning scenario where a model learns from a non-i.i.d. data stream, represented as $\mathcal{D}_1, ..., \mathcal{D}_T$, where $\mathcal{D}_t = \{(x_i^t, y_i^t)\}_{i=1}^{N_t}$ is the task-specific

subset, $x_i^t \in \mathbb{R}^{w \times h \times c}$ is an image input and $y_i^t \in \mathbb{Z}$ is its corresponding label. The goal of continual learning is to learn a function f_θ which maps the input x to the label y from an arbitrary task seen so far. We focus on two scenarios. In the class-incremental setting, each subset $\mathcal{D}_t$ contains a disjoint class label set. In the domain-incremental setting, the subsets $\mathcal{D}_1, \ldots, \mathcal{D}_T$ share the class labels, but the input distributions varies over time.

Nearest Mean Classifier (NMC) We decouple the goal of continual learning f_θ into two steps. The first step is to learn the representation h and the next is to learn the classifier g. We directly adopt a pre-trained vision transformer as our feature representation without training. For the classifier, inspired by [21, 17] we use the nearest mean classification strategy. During the training stage of task t, we calculate the mean features of a class in $\mathcal{D}_t$

$$\mu_k = \frac{1}{|C_k|} \sum_{x \in C_k} h(x) \ , \tag{1}$$

where C_k denotes the set of training samples belonging to class k. Only class mean features are saved in the memory and will be used during evaluation. At the test time of task t, the feature of a test sample is extracted by the pre-trained model, and the predicted class label is taken as the class whose mean features is the closest (over all the seen classes so far) to the feature of a test sample.

$$\hat{y} = \underset{k}{\operatorname{argmin}} ||h(x) - \mu_k|| \tag{2}$$

4 Experiments

We follow the experimental setup used in [27] to evaluate our method for a fair comparison. We test our method in class incremental learning, where new sets of classes are introduced to the model, and in domain incremental learning, where classes remain the same and the domain changes; see Sec 3.

Datasets: We evaluate our baseline on four common continual learning benchmarks, Split- CIFAR100 [11] , 5-datasets [6] and Split-ImageNet-R [8] in the class incremental learning setting. As proposed by [27] and [26]. Split-CIFAR-100 contains 10 tasks with 10 classes for each task. The 5-dataset benchmark concatenates 5 datasets, MNIST, SVHN, notMNIST , FashionMNIST and CIFAR10, with each dataset forming one task. Split ImageNet-R is a newly proposed dataset for continual learning by [26]. It consists of 200 classes which are randomly divided into 10 tasks. It contains the same object types however presented in different styles such as cartoon , graffiti and origami. These variations make the continual learning more challenging. For the domain incremental learning setting, we use CoRe50 proposed by [14]. It contains 50 objects collected in 11 distinct domains(tasks). 8 domains were faced and learned incrementally while the test is performed on the remaining three domains. Since a single test task is used, we do not report forgetting and joint training results in that scenario.

Evaluation Methods: For our approach (`Ours`), we employ the widely used ViT-B/16[5] model pre-trained on ImageNet-21k [24] provided by `timm` library [28]. We mainly compare our baseline with the recent L2P [27] which adopts the pretrained model as ours and learns a prompt pool with a prompt selection mechanism to modify the pretrained representations. We also consider popular continual learning methods including regularization-based methods (LwF[13] , EWC [10]) and rehearsal-based methods (ER[4], GDumb[19], BiC [29], DER++ [1] and Co^{2}L [3]). We present joint training results where the training data is, i.i.d. distributed among the whole benchmark with no task split. FT-frozen adds a fully-connected layer on top of the frozen feature extractor as the classification head, and FT allows end-to-end training on the feature extractor. Note that FT-frozen is different from our baseline, as we use NMC classifier and build it incrementally.

Results: Table 1, 4, and 2 report the performance of continual learning in incremental setting in Average Accuracy at the end of the learning sequence of Split-CIFAR100, Split-ImageNet-R, and the 5-dataset respectively. Table 4 shows the results of the continual learning in domain incremental setting on CoRe50[14]. The results are grouped based on the use of replay sample. Our simple baseline achieves the best performance on Split CIFAR-100 and CoRe50 while being the second best on Split-ImageNet-R and 5-dataset benchmarks. Our results on Split-CIFAR-100 are even better than methods that use replay samples. Concretely, our baseline achieves 88.53% with zero buffer size, which is 4.7% and 2.22% better than the runner-up and significantly more complicated L2P with no exemplars and L2P with 50 exemplars/class buffer, respectively. This suggests that pre-trained transformer offers a strong representation that achieves competitive performance. We think that the main reason for a possible inferior performance could be the ineffective design of the

Table 1: Continual learning performance expressed in Average Accuracy and Forgetting at the end of the learning sequence of CIFAR-100[11]. All methods are initialized with pretrained weights for a fair comparison. Our baseline outperforms all the methods in this benchmark.

Method	Buffer size	Average Acc	Forgetting
FT - frozen	0	17.72	59.09
FT	0	33.61	86.87
EWC[10]	0	47.01	33.27
LwF [13]	0	60.69	27.77
L2P [27]	0	83.83	7.63
Ours	0	**88.53**	-
ER [4]	50/class	82.53	16.46
GDumb [19]	50/class	81.67	-
BiC [29]	50/class	81.42	17.31
DER++ [1]	50/class	83.94	14.55
Co2L [3]	50/class	82.49	17.48
L2P [27]	50/class	86.31	5.83
Joint	-	90.85	-

Table 2: Continual learning performance expressed in Average Accuracy and Forgetting at the end of the learning sequence of the 5-dataset benchmark [6]. All methods are initialized with pretrained weights of the transformer for a fair comparison. Our baseline performs competitively with the exemplar-free methods in this benchmark

Method	Buffer size	Average Acc	Forgetting
FT - frozen	0	39.49	42.62
FT	0	20.12	94.63
EWC[10]	0	50.93	34.94
LwF [13]	0	47.91	38.01
L2P [27]	0	**81.14**	4.64
Ours	0	79.84	-
ER [4]	50/class	84.26	12.85
GDumb [19]	50/class	70.76	-
BiC [29]	50/class	85.53	10.27
DER++ [1]	50/class	84.88	10.46
Co2L [3]	50/class	86.05	12.28
L2P[27]	50/class	**88.95**	4.92
Joint		93.93	

Table 3: Continual learning performance in Average Accuracy at the evaluation task of CoRe50 [14]. All methods are initialized with pretrained weight for fair comparison

Method	Buffer size	Test Acc
EWC[10]	0	74.82
LwF[13]	0	75.45
L2P[27]	0	78.33
Ours	0	**83.23**
ER[4]	50/class	80.1
GDumb[19]	50/class	74.92
BiC[29]	50/class	79.28
DER++[1]	50/class	79.7
Co2L[3]	50/class	79.75
L2P[27]	50/class	81.07

Table 4: Continual learning performance in Average Accuracy and Forgetting at the end of the learning sequence of the Split-ImageNet-R[8] benchmark. All methods are initialized with pretrained weights for fair comparison.

Method	Buffer size	Average Acc.	Forgetting
FT - frozen	0	39.49	42.62
FT	0	28.87	63.80
EWC [10]	0	35.00	56.16
LwF [13]	0	38.54	52.37
L2P [27]	0	61.57	9.73
Ours	0	**65.99**	-
ER [4]	5000	65.18	23.31
GDumb [19]	5000	65.90	-
BiC [29]	5000	64.63	22.25
DER++ [1]	5000	**66.73**	20.67
Co2L [3]	5000	65.90	23.36
Joint	-	79.13	-

continual learning mechanisms compared to the robust features provided by the pretrained model. Such a model, pretrained on a large and diverse dataset, may have already captured most of the distribution properties in the evaluation benchmarks. Then a desired continual learning mechanism needs to further encourage the model to produce the task-invariant features specific to the deployed benchmark. Our nearest neighbor baseline is also competent among exemplar-free methods on Split-ImageNet-R and 5-datasets benchmarks, which is more diverse than CIFAR-100 and CoRe50. However, methods with higher buffer sizes and finely designed continual learning mechanisms do improve the representations extracted from pretrained models.

5 Conclusion

In this work, we explore the representational capacity of large-scale pre-trained models in continual learning settings. We provide simple nearest neighbor baseline experiments on four benchmarks, showing competitive performance to more sophisticated state-of-the-art continual learning methods which also leverage the same pretrained models. We agree that using pretrained weights can be a reasonable practice even in continual learning. However, to show real progress in continual learning systems, we need to focus more on building methods that can continually add quality to the learning representations. A desired continual learning algorithm shall go significantly beyond the knowledge embedded in the pretrained model. Another important aspect is the considered benchmarks for evaluating continual learning methods. Such benchmarks need to be sufficiently challenging and different from the data distributions employed for pretrained models.

References

[1] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and SIMONE CALDERARA. Dark Experience for General Continual Learning: a Strong, Simple Baseline. In *Advances in Neural Information Processing Systems*, volume 33, pages 15920–15930. Curran Associates, Inc., 2020.

[2] Francisco M. Castro, Manuel J. Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-End Incremental Learning. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, volume 11216, pages 241–257. Springer International Publishing, Cham, 2018. Series Title: Lecture Notes in Computer Science.

[3] Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2l: Contrastive continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision 2021*, pages 9516–9525, 2021.

[4] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K. Dokania, Philip H. S. Torr, and Marc'Aurelio Ranzato. On Tiny Episodic Memories in Continual Learning, June 2019. arXiv:1902.10486 [cs, stat].

[5] 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. In *International Conference on Learning Representations*, 2020.

[6] Sayna Ebrahimi, Franziska Meier, Roberto Calandra, Trevor Darrell, and Marcus Rohrbach. Adversarial continual learning. In *European Conference on Computer Vision*, pages 386–402. Springer, 2020.

[7] Jhair Gallardo. Self-Supervised Training Enhances Online Continual Learning. page 15.

[8] 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*, pages 8340–8349, 2021.

[9] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a Unified Classifier Incrementally via Rebalancing. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 831–839, Long Beach, CA, USA, June 2019. IEEE.

[10] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, March 2017. Publisher: Proceedings of the National Academy of Sciences.

[11] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009.

[12] Timothée Lesort, Oleksiy Ostapenko, Diganta Misra, Md Rifat Arefin, Pau Rodríguez, Laurent Charlin, and Irina Rish. Scaling the Number of Tasks in Continual Learning, July 2022. arXiv:2207.04543 [cs].

[13] Zhizhong Li and Derek Hoiem. Learning without Forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, December 2018. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[14] Vincenzo Lomonaco and Davide Maltoni. Core50: a new dataset and benchmark for continuous object recognition. In *Conference on Robot Learning*, pages 17–26. PMLR, 2017.

[15] Michael McCloskey and Neal J. Cohen. Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. In Gordon H. Bower, editor, *Psychology of Learning and Motivation*, volume 24, pages 109–165. Academic Press, January 1989.

[16] Michael McCloskey and Neal J. Cohen. Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. In Gordon H. Bower, editor, *Psychology of Learning and Motivation*, volume 24, pages 109–165. Academic Press, January 1989.

[17] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Distance-Based Image Classification: Generalizing to New Classes at Near-Zero Cost. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2624–2637, November 2013.

[18] Oleksiy Ostapenko, Timothee Lesort, Pau Rodríguez, Md Rifat Arefin, Arthur Douillard, Irina Rish, and Laurent Charlin. Continual Learning with Foundation Models: An Empirical Study of Latent Replay, July 2022. arXiv:2205.00329 [cs].

[19] Ameya Prabhu, Philip H. S. Torr, and Puneet K. Dokania. GDumb: A Simple Approach that Questions Our Progress in Continual Learning. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, volume 12347, pages 524–540. Springer International Publishing, Cham, 2020. Series Title: Lecture Notes in Computer Science.

[20] 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*, pages 8748–8763. PMLR, 2021.

[21] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. iCaRL: Incremental Classifier and Representation Learning, April 2017. arXiv:1611.07725 [cs, stat].

[22] Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. ImageNet-21K Pretraining for the Masses. December 2021.

[23] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, December 2015.

[24] Andreas Peter Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *Transactions on Machine Learning Research*, 2022.

[25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems 2017*, 30, 2017.

[26] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. DualPrompt: Complementary Prompting for Rehearsal-free Continual Learning, August 2022. arXiv:2204.04799 [cs].

[27] 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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, March 2022.

[28] Ross Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019.

[29] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large Scale Incremental Learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 374–382, Long Beach, CA, USA, June 2019. IEEE.