Probing Representation Forgetting in Continual Learning

MohammadReza Davari, Eugene Belilovsky
Concordia University and Mila
{mohammadreza.davari,eugene.belilovsky}@concordia.ca

Abstract

Continual Learning methods typically focus on tackling the phenomenon of catastrophic forgetting in the context of neural networks. Catastrophic forgetting is associated with an abrupt loss of knowledge previously learned by a model. In supervised learning problems this forgetting is typically measured or observed by evaluating decrease in task performance. However, a model's representations can change without losing knowledge. In this work we consider the concept of representation forgetting, which relies on using the difference in performance of an optimal linear classifier before and after a new task is introduced. Using this tool we revisit a number of standard continual learning benchmarks and observe that through this lens, model representations trained without any special control for forgetting often experience minimal representation forgetting. Furthermore we find that many approaches to continual learning that aim to resolve the catastrophic forgetting problem do not improve the representation forgetting upon the usefulness of the representation.

1 Introduction

Continual Learning (CL) is concerned with developing methods for learners to manage changing distributions. The goal being to acquire new knowledge from new data distributions while avoiding forgetting of previous knowledge. A common scenario is CL in the classification setting, where the class labels presented to the learner change over time. In this scenario a phenomenon known as catastrophic forgetting has been commonly observed [13, 22]. This phenomenon is often described as a loss of knowledge about previously seen data and observed in the classification setting as a decrease in accuracy.

Deep Learning has been traditionally motivated as an approach which can automatically learn representations [7], forgoing the need to design handcrafted features for data. Indeed representation learning is at the core of deep learning methods in supervised and unsupervised settings [12]. In the case of many practical scenarios we may not be simply interested in the final performance of the model, but also the usefulness of its features for various downstream tasks [30]. Although a representation may change drastically at task boundaries [8], this does not necessarily entail a loss of useful information and may instead correspond to a simple transformation. For example consider a standard multi-head CL setting, where each task shares a representation and only differs through task heads. A permutation of features leads to total catastrophic forgetting as measured by standard approaches as the task heads no longer match with the representations, but this does not correspond to a loss of knowledge about the data.

As CL envisions having learners operate over long time horizons while continually maintaining old knowledge and integrating new information, it is sensible to consider the usefulness of their representations for previous tasks in addition to directly measuring the performance on previous tasks using the last layer classifiers. In this paper we highlight that traditional approaches of evaluating forgetting are unable to properly disambiguate trivial changes in the features (e.g. permutation) from

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

abrupt losses of useful representations. We propose to use linear probes, commonly used to study unsupervised representations [11] and intermediate layer representations [25, 33], to evaluate CL algorithms and their usefulness. In this work we revisit several classical CL settings and benchmarks and attempt to measure how much forgetting is observed by the representations using optimal Linear Probes (LP). We observe that in many commonly studied cases of catastrophic forgetting the representations can be observed to avoid losing critical task information.

2 Related Work

Multiple approaches have been developed for CL, often the design of these methods is focused on mitigating the catastrophic forgetting phenomenon, with aspects such as maximizing forward and backward transfer between tasks taken as secondary [20]. One class of methods focuses on bypassing this problem by growing architectures over time as new tasks arrive [2, 17, 29, 31]. Under the fixed architecture setting, one can identify two primary directions. In the first, methods rely on storing and re-using samples from the previous history while learning new ones, this includes approaches such as GEM [20] and ER [10]. The second class of methods encode the knowledge of the previous tasks in a prior that is used to regularize the training of the new task and includes approaches such as [1, 15, 19, 24, 34]. A classic method in this vain is Learning without Forgetting (LwF) [18], which mitigates forgetting by a regularization term that distills knowledge [14] from the earlier tasks. Prior to learning a new task, the network representations are recorded, and are used during the training to regularize the objective by distilling knowledge from the earlier state of the network. In Section 4 we will examine the effectiveness of this approach in mitigating representation forgetting.

Recent works on elucidating the nature of catastrophic forgetting have examined the influence of task sequence [23], network architecture [6], and change in representation similarity [27]. Our work is related in spirit to [27] as we pursue measuring how much forgetting has occurred on the learned representation and we additionally study this for depth. However, in [4], the authors use linear CKA [16] to measure the similarity between representations influenced by forgetting, while in our work we measure how much forgetting is observed by the representations using LP.

Several works [21] have focused on modifying the last layer of a classification network to make more effective use of the representation for prior tasks. This indirectly highlights that the last layer can be modified to yield better performance on prior tasks. Particularly [21, 28] use a buffer of old examples at evaluation time to construct a class mean prototype. This allows to more effectively use the representation of the network. These works, however consider settings where CL methods are used to control training, while we also emphasize naive continuation of training under task shift can also yield strong representations. Our work can also be seen as a way to explain and motivate the need for such approaches.

3 Linear Probes

Following work in supervised learning [11] and in the analysis of intermediate representations [33] we evaluate the usefulness of representations by an optimal linear classifier using training data from the original task. A linear classifier is trained on top of the frozen activations of the base network given the training instances of a particular dataset. The test set accuracy obtained by LP on the aforementioned dataset is used as a proxy to measure the quality of the representations. The difference in performance of the LP before and after a new task is introduced acts as a surrogate measure to the amount of forgetting observed by the representations and is referred to as representation forgetting.

4 Experiments

We perform evaluations in several published CL scenarios, focusing on the task-incremental setting. We first consider the setup from [5, 27]. Subsequently we revisit the evaluations of [18] and finally we consider a longer task sequence in the online and offline setting with different model capacity [20]. In all cases LP are trained to convergence with Adam and a learning rate of 1e-3.

Two Task SplitCIFAR10 Sequence We consider a two task SplitCIFAR10 setting from [27]. We use the same models and training procedures and subsequently evaluate the forgetting experienced by the representations. In Table 1, we study the shift in representations of each block of the network by measuring the performance of LP on Task 1 data before and after training the network on Task 2.

First observe that the model accuracy decreases from 85% to 63% suggesting a large degradation in performance and a large forgetting. However, following the optimal classifier evaluation protocol

Table 1: Representation forgetting of Task 1 measured via optimal linear probes (LP) on ResNet and VGG. The Accuracy degradation of LP trained on activations of stages (blocks of convolutions) before and after observing Task 2 suggests that the representations are still highly useful for Task 1 despite training on Task 2.

ResNet: Network accuracy on task-1 after task-2 training: 63.64%			
Block	LP Acc. After Task-1	LP Acc. After Task-2	Δ Acc.
Block-0	63.54%	64.62%	+1.08%
Block-1	68.24%	69.50%	+1.26%
Block-2	71.62%	71.34%	−0.28%
Block-3	77.64%	76.52%	−1.12%
Block-4	80.06%	78.98%	−1.08%
Block-5	85.82%	80.10%	−5.72%
Block-6	85.94%	79.12%	−6.82%
VGG: Network accuracy on task-1 after task-2 training: 57.88%			
Block-0	67.94%	66.86%	−1.08%
Block-1	73.60%	72.52%	−1.08%
Block-2	78.58%	75.68%	−2.90%
Block-3	81.54%	75.48%	−6.06%

Table 2: Forgetting of Task 1 measured via optimal linear probes (LP). Note that although the forgetting is much higher for fine-tuning compared to LwF, the LP accuracy is nearly identical, especially for the ImageNet → CUB task, suggesting that LwF does not improve over naive fine-tuning in terms of forgetting knowledge acquired on ImageNet.

Network Acc. on ImageNet: 71.59%				
	ImageNet (T-1) → CUB (T-2)		ImageNet (T-1) → Scenes (T-2)	
	LP Acc. After T-2	T-1 Acc. After T-2	LP Acc. After T-2	T-1 Acc. After T-2
Fine-tune	61.12%	51.12%	65.81%	63.96%
LwF	61.16%	61.02%	66.16%	67.66%

the accuracy degradation is observed to be only 6.8%, without any CL method applied to control forgetting. This suggests the representations are still highly useful for Task 1 despite training on Task 2. Second, similar to [27] we observe the forgetting is concentrated at the top layers. Indeed early layers in the network experience almost no representation forgetting and in some cases improve their usefulness with regards to Task 1. [27]'s analysis also showed forgetting occurring in early layers to a lower degree than in higher layers and suggested that forgetting is extreme in the upper layer representations. Specifically, the authors measured linear CKA [16] performance between layers (Fig. 1 in [27]) showing this similarity metric dropped progressively from close to 1 to 0.2 for both ResNet and VGG models. However, our evaluation suggest forgetting doesn't exist in lower layers and the loss in information is less catastrophic at higher layers than suggested by [27].

ImageNet Transfer We now move to larger scale scenarios of models trained on large datasets and applied to a different task. We take the setting of [18], which considers the ImageNet [30] transfer to various datasets, in particular CUB [32] and Scenes [26]. We use the same model (VGG-16) and training procedures described in [18]. The LwF method applies a regularizer to the training objective by distilling knowledge from the earlier state of the network, which constrains the optimization space of the parameters for the new task. Table 2 shows the results in this setting which are our reproduction of [18] and additionally perform the LP evaluation using ImageNet data on Task 2 models. Note that the LP training does not use any data augmentations. Our evaluation reveals that although the traditional forgetting is much higher for fine-tuning compared to LwF, the LP accuracy is nearly identical, especially for the CUB transfer task. This suggests that LwF does not improve over naive fine-tuning in terms of forgetting knowledge acquired on ImageNet.

Longer Task Sequences and Variable Model Capacity So far we have studied two task sequence. We now consider the SplitCIFAR10 benchmark popularly used in a variety of CL work [3, 9, 28] that contains a 5 task sequence. We train models using a 5 task sequence and a multi-head setting. We then evaluate a LP trained on all the data to compare the optimal classifier performance across

Table 3: Final Accuracy of 5 task SplitCIFAR10 Sequence with Variable Width for online and offline training. M indicates the number of samples per task used in the ER buffer. We observe that simple finetuning baseline shows large forgetting which does not seem to improve with width, and further degrades with increased training time per task (online vs offline). On the other hand LP evaluation reveals that representation quality for finetuning becomes closer to strong CL methods such as ER.

		Resnet10, Width=20		Resnet10, Width=100	
		Observed Acc.	LP Acc.	Observed Acc.	LP Acc.
online	Finetune	71.4%	83.1%	72.4%	88.0%
	ER-M5	81.2%	85.2%	84.2%	90.6%
	ER-M20	82.8%	86.4%	86.8%	90.8%
offline	Finetune	65.8%	83.6%	60.0%	87.8%
	ER-M5	84.6%	88.8%	87.6%	92.2%
	ER-M20	89.2%	90.8%	89.8%	92.8%

methods and across model capacity. We consider both the online setting where the data samples are seen only once as well as the offline setting where the learner receives the entire set of task data and is allowed to train for 10 epochs. In all cases we train with SGD and a learning rate of 0.01.

A number of recent works have illustrated that Experience Replay, particularly as the buffer size increases, is a strong baseline [10, 27] in this setting. Thus we use Experience replay with both a small buffer, M=5 samples per class, and a relatively large buffer, M=20 samples per class, to allow a representative comparison of fine-tuning and popular CL methods. To simplify the analysis we report the final accuracy averaged on all tasks observed after the task sequence and the accuracy of a LP trained on all the training data. Table 3 shows the results in both the online and offline setting for two different models. One model is the modified Resent18 [20] with a width parameter of 20 used commonly in [3, 20] and the other is the same network but with all layers widened by a factor of 5.

First, we observe that as in the other cases the LP accuracy of fine-tuning is higher than the observed accuracy, suggesting forgetting is less catastrophic than suggested by observed accuracy. Secondly, we observe that the fine-tuning evaluated using the observed accuracy is particularly deceptive in revealing how the model representations change both from online to offline case and especially with increasing capacity. Using observed accuracy one would conclude that increasing width and capacity of the model without applying any CL specific method does not improve performance and can even decrease model performance overall due to forgetting. This is consistent with the Appendix of [5], which evaluates only on observed accuracy. However, if we observe the LP accuracy, it reveals a more clear picture of what occurs at the representation level, suggesting that larger models can indeed reduce forgetting even when trained from scratch without explicit control of forgetting. Moreover we observe that at the representation level as model capacity increases, naive fine-tuning becomes much closer in performance to costly (and under privacy constraints unusable) CL methods such as ER which use more computation and memory.

Contrary to the observations of [5] our results illustrate that the model capacity does play a profound importance on forgetting even when the model is trained from scratch. That is the observed overall accuracy at the end of the sequence does not greatly increase as the model widens, while the LP accuracy does greatly increase and yields representations that are much closer for fine-tuning and methods which explicitly combat forgetting, some with large buffers.

5 Conclusion

We have highlighted the importance of evaluating representations and not just task accuracy in CL settings. Our results suggest a) the feature forgetting under naive training in supervised settings is not as catastrophic as other metrics suggest b) we reconfirm that forgetting is concentrated at the top layers and show that forward transfer can happen in lower layers under naive (non-CL specific) training and c) We demonstrate that without evaluation of features the effects of model size on forgetting and representation learning will be misinterpreted.

6 Acknowledgements

This work is supported by NSERC Discovery Grant "Towards Continual Learning in the Visual World" and GPU computation from Compute Canada.

References

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *ECCV 2018*.

[2] Rahaf Aljundi, Punarjay Chakravarty, and Tinne Tuytelaars. Expert gate: Lifelong learning with a network of experts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[3] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. *arXiv preprint arXiv:1903.08671*, 2019.

[4] Ankesh Anand, Evan Racah, Sherjil Ozair, Yoshua Bengio, Marc-Alexandre Côté, and R Devon Hjelm. Unsupervised state representation learning in atari. *arXiv preprint arXiv:1906.08226*, 2019.

[5] Anonymous. Effect of scale on catastrophic forgetting in neural networks. In *Submitted to The Tenth International Conference on Learning Representations*, 2022. under review.

[6] Gaurav Arora, Afshin Rahimi, and Timothy Baldwin. Does an LSTM forget more than a CNN? an empirical study of catastrophic forgetting in NLP. In *Proceedings of the The 17th Annual Workshop of the Australasian Language Technology Association*, pages 77–86, Sydney, Australia, 4–6 December 2019. Australasian Language Technology Association.

[7] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[8] Lucas Caccia, Rahaf Aljundi, Tinne Tuytelaars, Joelle Pineau, and Eugene Belilovsky. Reducing representation drift in online continual learning. *arXiv preprint arXiv:2104.05025*, 2021.

[9] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. *arXiv preprint arXiv:1801.10112*, 2018.

[10] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. Continual learning with tiny episodic memories. *arXiv preprint arXiv:1902.10486*, 2019.

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep learning. Book in preparation for MIT Press, 2016.

[13] Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*, 2013.

[14] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[15] 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. *arXiv preprint arXiv:1612.00796*, 2016.

[16] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pages 3519–3529. PMLR, 2019.

[17] Xilai Li, Yingbo Zhou, Tianfu Wu, Richard Socher, and Caiming Xiong. Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting. In *International Conference on Machine Learning*, pages 3925–3934. PMLR, 2019.

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

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

[20] David Lopez-Paz et al. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, pages 6467–6476, 2017.

[21] Zheda Mai, Ruiwen Li, Hyunwoo Kim, and Scott Sanner. Supervised contrastive replay: Revisiting the nearest class mean classifier in online class-incremental continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3589–3599, 2021.

[22] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of learning and motivation*, 24:109–165, 1989.

[23] Cuong V Nguyen, Alessandro Achille, Michael Lam, Tal Hassner, Vijay Mahadevan, and Stefano Soatto. Toward understanding catastrophic forgetting in continual learning. *arXiv preprint arXiv:1908.01091*, 2019.

[24] Cuong V Nguyen, Yingzhen Li, Thang D Bui, and Richard E Turner. Variational continual learning. *arXiv preprint arXiv:1710.10628*, 2017.

[25] Edouard Oyallon. Building a regular decision boundary with deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5106–5114, 2017.

[26] Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 413–420. IEEE, 2009.

[27] Vinay V Ramasesh, Ethan Dyer, and Maithra Raghu. Anatomy of catastrophic forgetting: Hidden representations and task semantics. *arXiv preprint arXiv:2007.07400*, 2020.

[28] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proc. CVPR*, 2017.

[29] Amir Rosenfeld and John K Tsotsos. Incremental learning through deep adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 42(3):651–663, 2018.

[30] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[31] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

[32] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[33] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[34] Friedemann Zenke, Ben Poole, and Surya Ganguli. Improved multitask learning through synaptic intelligence. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2017.

APPENDIX

Reproducing LwF Results

We followed the training procedure as closely as possible to the ones reported by [19]. However, our results are slightly different from the ones reported due to variations. Table 4 highlights these differences.

Table 4: Forgetting of task-1 measured via probing networks.

	ImageNet (T-1) → CUB (T-2)			ImageNet (T-1) → Scenes (T-2)		
	T1 Acc.	T2 Acc.	T-1 Acc. After T-2	T1 Acc.	T2 Acc.	T-1 Acc. After T-2
Finetune-[19]	68.6%	73.1%	50.7%	68.6%	74.6%	62.7%
Finetune-Ours	71.6%	75.0%	51.1%	71.6%	77.1%	64.0%