Discriminative Representation Loss (DRL) for Continual Learning: A Solution for CLVision Challenge

Yu Chen
University of Bristol
United Kingdom
yc14600@bristol.ac.uk

Jian Ma
University of Bristol
United Kingdom
jian.ma@bristol.ac.uk

HanYuan Wang
University of Bristol
United Kingdom
hanyuan.wang@bristol.ac.uk

YuHang Ming
University of Bristol
United Kingdom
yuhang.ming@bristol.ac.uk

Jordan Massiah
Amazon
United Kingdom
jormas@amazon.com

Tom Diethe
Amazon
United Kingdom
tdiethe@amazon.com

Abstract

The use of episodic memories in continual learning is an efficient way to prevent the phenomenon of catastrophic forgetting. *In recent studies, several gradient-based approaches have been developed to make more efficient use of compact episodic memories, which constrain the gradients resulting from new samples with gradients from memorized samples. In this paper, we propose a method for decreasing the diversity of gradients through an extra optimization objective that we call Discriminative Representation Loss, instead of directly re-projecting the gradients. Our methods show promising performance with relatively cheap computational cost on all the three tracks of the CLVision challenge* [1].

1. Introduction

Continual learning is to enable a model to sequentially learn tasks, imitating the way that humans are able to learn new tasks without forgetting previously learned ones, using common knowledge shared across different skills. The fundamental problem in continual learning is *catastrophic forgetting* [12, 7], *i.e.* (neural network) models tend to forget previously learned tasks while learning new ones.

There are two main categories to alleviating forgetting in continual learning: *i*) preserving parameters of previous tasks, including methods for parameter regularization [7, 21, 14] and methods for incrementally evolving the model [18, 6]; *ii*) preserving the knowledge of data distributions of previous tasks, including replay-based methods [19, 16], methods for generating compact episodic memories [4, 1], and methods using episodic memories to refine gradients when updating model parameters [11, 3, 15].

Gradient-based approaches using episodic memories, in particular, have been receiving increasing attention. The essential idea is to use gradients produced by samples from episodic memories to constrain the gradients produced by new samples, *e.g.* by ensuring the inner product of the pair of gradients is non-negative [11] as follows:

$$\langle g_t, g_k \rangle = \left\langle \frac{\partial \mathcal{L}(x_t, \theta)}{\partial \theta}, \frac{\partial \mathcal{L}(x_k, \theta)}{\partial \theta} \right\rangle \geq 0, \forall k < t \quad (1)$$

where t and k are time indices, x_t denotes a new sample from the current task, and x_k denotes a sample from the episodic memory. Thus, the updates of parameters are forced to preserve the performance on previous tasks as much as possible. In Gradient Episodic Memory (GEM) [11], g_t is projected to a direction that closest to it in L_2-norm whilst also satisfying Eq. (1):

$$\min_{\tilde{g}} \frac{1}{2} ||g_t - \tilde{g}||_2^2, \;\; s.t. \langle \tilde{g}, g_k \rangle \geq 0, \;\; \forall k < t \quad (2)$$

Optimization of this objective requires a high-dimensional quadratic program and thus is computationally expensive. Averaged-GEM (A-GEM) [2] alleviates the computational burden of GEM by using the averaged gradient over a batch of samples instead of individual gradients of samples in the episodic memory. This not only simplifies the computation, but since there are fewer constraints, also obtains better performance than GEM. [1] propose Gradient-based Sample Selection (GSS), which selects samples that produce gradients with maximum diversity to store in episodic memory. Here diversity is measured by the cosine similarity between gradients. Since the cosine similarity is computed using the

[1] https://competitions.codalab.org/competitions/23317

inner product of two normalized gradients, GSS embodies the same principle as other gradient-based approaches with episodic memories. Although GSS suggests the samples with most diverse gradients are important to the generalization across tasks, [3] show that the average gradient over a small set of random samples may be able to obtain good generalization as well.

In this paper, we demonstrate that the diversity of gradients correlates to the diversity of representations. Accordingly, we propose a new objective, Discriminative Representation Loss (DRL), for classification tasks in online continual learning. Our methods show promising performance with relatively low computational cost across all three tasks in the Continual Learning in Computer Vision (CLVision) challenge.

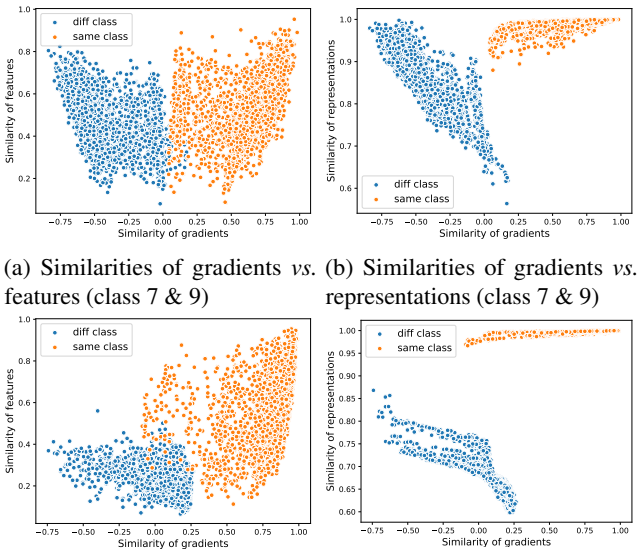
2. Discriminative Representation Loss

According to Eq. (1), larger cosine similarities between gradients produced by current and previous tasks result in improved generalisation. This in turn indicates that samples that lead to the most diverse gradients provide the most difficulty during learning. This can be interpreted from the perspective of constrained optimization as discussed in [1]. Moreover, the diversity of gradients relates to the Gradient Signal to Noise Ratio (GSNR) [9], which plays a crucial role in the model's generalization ability:

$$\text{GSNR} = \mathbb{E}^2[g]/\text{Var}[g], \quad g \sim P(\nabla_\theta \mathcal{L}(x, \theta)), \quad x \sim \mathbb{D},$$

where $\mathbb{D}$ denotes the data distribution. Intuitively, when more of the gradients point in the same direction, the variance will be smaller, and the mean will be larger, leading to a larger GSNR, and consequently, improved test-time performance. In this sense, reducing the diversity of gradients may improve the generalization ability of models.

We show that the diversity of gradients correlates to diversity of representations by experiments with MNIST [8] dataset. We first trained two binary classifiers for two groups of MNIST classes ($\{0, 1\}$ and $\{7, 9\}$). The classifiers have two hidden layers each with 100 hidden units. We randomly chose 100 test samples from each group, and computed the pairwise cosine similarities of gradients, features and representations. Features are raw input representations of the data, i.e. 784-dimensional pixels of MNIST data. Representations are obtained by concatenating the output of all layers of the neural network, including the logits layer. We display the different similarities in Fig. 1, where blue dots indicate the similarity between two samples from two different classes, while orange dots indicate that the two samples are from the same class. In Figs. 1a and 1c, the correlation coefficients of blue dots are -0.37 and -0.38, which of orange dots are 0.38 and 0.36. In Figs. 1b and 1d, the correlation coefficients of blue dots are -0.86 and -0.85, which of orange dots



(a) Similarities of gradients *vs.* features (class 7 & 9)

(b) Similarities of gradients *vs.* representations (class 7 & 9)

(c) Similarities of gradients *vs.* features (class 0 & 1)

(d) Similarities of gradients *vs.* representations (class 0 & 1)

Figure 1: Similarities of gradients, features and representations of two classes in MNIST dataset. The x axis is the cosine similarity of gradients, the y axis is the cosine similarity of features in (a) and (c), and representations in (b) and (d). Blue dots indicate the similarity between two samples from two different classes, while orange dots indicate that the two samples are from the same class.

are 0.71 and 0.79. In all cases, the similarities of representations show stronger correlations with the similarities of gradients than those of features. This is especially true when the compared samples are from different classes (blue dots in Fig. 1): here larger similarities of representations correspond to smaller similarities of gradients. In addition, the blue and orange dots are perfectly separable on the y axis in Fig. 1d, which indicates that the classifier for class 0 and 1 has learnt strongly discriminative representations, and as a result achieves nearly perfect (99.95%) accuracy on the test set. In comparison, the classifier for class 7 and 9 has learnt less discriminative representations, resulting in lower test accuracy (96.25%).

The results show that the discrimination ability of representations strongly correlates with the diversity of gradients, and more discriminative representations lead to more consistent gradients. We use this insight to introduce an extra objective Discriminative Representation Loss (DRL) into the optimization objective of classification tasks in continual learning. Instead of explicitly refining gradients during training process, DRL helps with decreasing gradient diversity by optimizing the representations. As defined in Sec. 2, DRL consists of two parts: one is for minimizing the similarities of representations between samples from different classes ($\mathcal{L}_{bt}$), the other is for maximizing the similarities of

representations between samples from a same class ($\mathcal{L}_{wi}$).

$$\min_{\Theta} \mathcal{L}_{DR} = \min_{\Theta}(\mathcal{L}_{bt} - \mathcal{L}_{wi}),$$

$$\mathcal{L}_{bt} = \frac{1}{B_{bt}} \sum_{l=1}^{L} \sum_{i=1}^{B} \sum_{j=1, y_j \neq y_i}^{B} \langle h_{l,i}, h_{l,j} \rangle,$$

$$\mathcal{L}_{wi} = \frac{1}{B_{wi}} \sum_{l=1}^{L} \sum_{i=1}^{B} \sum_{j=1, j \neq i, y_j = y_i}^{B} \langle h_{l,i}, h_{l,j} \rangle, \quad (3)$$

where Θ denotes the parameters of the model, L is the number of layers of the model, B is training batch size. B_{bt} and B_{wi} denote the number of pairs of samples in the training batch that are from different classes and the same class, respectively, $h_{l,i}$ is the output of layer l by input x_i and y_i is the label of x_i. We omit the normalization terms in cosine similarity because they do not improve performance while increasing computation cost. Essentially, DRL optimizes large margins between classes in a transformed feature space which is analogous to Kernel Fisher Discriminant analysis (KFD) [13] and distance metric learning [20] but with following major differences: *i*) in DRL the transformation function is explicit (e.g. the neural network) and the objective is w.r.t. its parameters, whereas it is implicit in KFD and metric learning; *ii*) DRL is computed by a batch of samples instead of the whole dataset, and it is possible to have $\mathcal{L}_{wi} = 0$ when there are no more than one sample from a given class in that batch. In a summary, DRL bears some resemblance to KFD and metric learning, but is tailored to neural networks in the (online) continual learning setting.

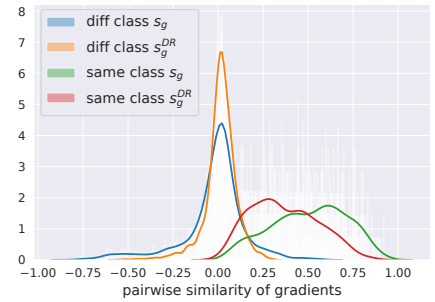
The final loss function combines a classification loss ($\mathcal{L}_{clf}$) with DRL, where we select $\mathcal{L}_{clf}$ to be the commonly used cross entropy loss for classification tasks in our experiments:

$$\mathcal{L} = \mathcal{L}_{clf} + \lambda \mathcal{L}_{DR}, \quad \lambda > 0, \quad (4)$$

where λ is a hyperparameter controlling the strength of $\mathcal{L}_{DR}$, which is larger for increased resistance to forgetting, and smaller for greater elasticity. We observe that relatively larger λ results in improved performance for more homogeneous tasks, as there is reduced conflict between forgetting and elasticity in such cases.

To verify if this objective can reduce the diversity of gradients, we compare the distributions of similarities of gradients and representations with/without DRL by training a model on Disjoint MNIST tasks. Fig. 2a shows the changes to similarities of representations caused by DRL, the similarities from different classes notably shifting towards a smaller region, as expected. Interestingly, similarities from the same class also shift slightly towards a smaller region. Accordingly, the similarities of gradients from different classes are more concentrated around 0, whilst the similarities from

(a) Comparing similarities of representations



(b) Comparing similarities of gradients

Figure 2: Distributions of similarities of gradients and representations with/without DRL. s_h^{DR} and s_h denote similarities of representations with and without DRL, respectively, s_g^{DR} and s_g denote similarities of gradients with and without DRL, respectively.

the same class are shifted towards 0 (Fig. 2b). We interpret this as the generalization between different classes can also prevent overfitting of individual classes.

3. Replay Strategies

There are different replay strategies for utilising samples in the episodic memory. The most basic one is to shuffle the memorized samples with new training data and train the model as usual. Experience Replay (ER) [3] suggests composing a training batch divided equally between samples from the episodic memory and samples from the current task. Since DRL depends on the pairwise similarities of samples in the training batch, we would prefer the training batch to include as wide a variety of different classes as possible to obtain sufficient discriminative information. Hence, we adjust the ER algorithm for the needs of DRL. The basic idea is to uniformly sample from all tasks (classes) in the memory buffer to form a training batch, so that this batch is evenly distributed across all seen tasks (classes). We call this Balanced Experience Replay (BER), details are in Alg. 1.

4. Experiments

We conducted experiments on all the three tracks of CLVision challenge which are based on the CORe50 [10]

Algorithm 1 Balanced Experience Replay

Input: $\mathcal{M}$ - memory buffer, $\mathcal{C}$ - the set of tasks in $\mathcal{M}$, $\mathcal{M}_c$ - samples of task c in $\mathcal{M}$, B - batch size, Θ - model parameters, $\mathcal{L}_\Theta$ - loss function w.r.t. Θ.
$\mathcal{B}_{train} = \emptyset$
for c **in** $\mathcal{C}$ **do**
  $\mathcal{B}_c \overset{B}{\sim} \mathcal{M}_c$ $\quad \triangleleft$ sample B samples from $\mathcal{M}_c$
  $\mathcal{B}_{train} = \mathcal{B}_{train} \cup \mathcal{B}_c$
end for
$\Theta \leftarrow$ Optimizer$(\mathcal{B}_{train}, \Theta, \mathcal{L}_\Theta)$

dataset:

1. New Instances (NI): In this setting 8 training batches of the same 50 classes are encountered over time. Each training batch is composed of different images collected in different environmental conditions.

2. Multi-Task New Classes (Multi-Task-NC): In this setting the 50 different classes are split into 9 different tasks: 10 classes in the first batch and 5 classes in the other 8. In this case the task label will be provided during training and test.

3. New Instances and Classes (NIC): this protocol is composed of 391 training batches containing 300 images of a single class. No task label will be provided and each batch may contain images of a class seen before as well as a completely new class.

For all the experiments, we use SGD optimizer with 0.01 learning rate, epoch is set to 1, our tests are based on ResNet50 [5] and ResNeSt50 [22] which are pre-trained on ImageNet [17]. Tabs. 1 to 3 show the results of each track of the challenge. The test accuracy of some tests are from submissions we submitted to the challenge and those without test accuracy are results we have not submitted.

We have tested three different replay strategies as introduced in Sec. 3. BER gives higher accuracy than ER in NI track, whereas ER gets better accuracy than BER in Multi-Task-NC track. The difference is the former uses a single-head model and the latter uses a multi-head one. Interestingly, the basic shuffle strategy works better for NIC. It is probably because the task sequence is very long (391 tasks) and the training set is quite small (300 samples) for each task. In such a case, the memorised samples from previous tasks will compose the majority of the training samples in latter tasks. By the shuffle strategy all the memorised samples will be trained on for certain whereas ER and BER may missed some samples during training due to the random sampling from the memory. In general, ResNeSt50 works better than ResNet50 and applying DRL obtains better performance than without it.

5. Discussion and conclusion

The two fundamental problems of continual learning with small episodic memories are how to: *(i)* make the best use of a small set of samples; and *(ii)* construct a small set of samples that are most representative of a large dataset. Gradient based approaches have shown that the diversity of gradients computed on data from different tasks is a key to generalization over these tasks. In this paper, we connect the diversity of gradients to discrimination ability of representations learned by the model, which leads to an alternative way to reduce the diversity of gradients instead of re-projecting gradients directly.

Our methods provide an approach to solving the first problem: more discriminative representations result in better generalization over different classes. This aligns with general classification tasks but under more restrictive conditions as found in the setting of continual learning. For the second question, we will leave it as a future work.

References

[1] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. In *Advances in Neural Information Processing Systems*, pages 11816–11825, 2019.

[2] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-GEM. In *International Conference on Learning Representations*, 2019.

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

[4] Yu Chen, Tom Diethe, and Neil Lawrence. Facilitating bayesian continual learning by natural gradients and stein gradients. *Continual Learning Workshop of 32nd Conference on Neural Information Processing Systems (NeurIPS 2018)*, 2018.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[6] Ching-Yi Hung, Cheng-Hao Tu, Cheng-En Wu, Chien-Hung Chen, Yi-Ming Chan, and Chu-Song Chen. Compacting, picking and growing for unforgetting continual learning. In *Advances in Neural Information Processing Systems*, pages 13647–13657, 2019.

[7] 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*, page 201611835, 2017.

[8] Yann LeCun, Corinna Cortes, and Christopher JC Burges. MNIST handwritten digit database. *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist*, 2, 2010.

Table 1: Experiment results of NI

	Episodic Memory	Batch Size	ER type	λ_{DR}	Test Acc.	Avg. Valid Acc.	RAM Usage (Mb)	Time (m)
ResNet50	100	32	Shuffle	0.	0.79	0.74	19678.42	6.
	100	16	Shuffle	0.	N/A	0.75	16573.91	8.62
	100	16	ER	0.	N/A	0.76	16560.55	9.92
	100	16	BER	0.	N/A	0.75	16611.49	19.71
	100	16	ER	0.0002	0.81	0.79	16558.13	10.1
	100	16	BER	0.0002	0.82	0.78	16625.19	19.88
	300	16	BER	0.0002	0.83	0.77	16447.77	19.56
	300	16	BER	0.001	0.83	0.79	16502.56	19.88
	2000	16	BER	0.001	N/A	0.77	17506.92	20.48
ResNeSt50	1500	16	BER	0.	N/A	0.77	18340.47	40.95
	1500	16	BER	0.001	0.89	0.81	18342.86	21.25

Table 2: Experiment results of Multi-Task-NC

	Episodic Memory	Batch Size	ER type	λ_{DR}	Test Acc.	Avg. Valid Acc.	RAM Usage (Mb)	Time (m)
ResNet50	20	32	Shuffle	0.	0.9	0.51	23863.71	5.66
	100	32	Shuffle	0.	N/A	0.50	24121.09	6.14
	100	16	Shuffle	0.	N/A	0.52	19400	8.73
	100	16	BER	0.0002	0.94	0.54	19583.7	20.92
	100	16	ER	0.0002	N/A	0.53	19579	10.0
	500	16	ER	0.0002	0.95	0.54	19047.21	9.7
ResNeSt50	500	16	ER	0.0002	0.97	0.54	19167.67	13.47

Table 3: Experiment results of NIC

	Episodic Memory	Batch Size	ER type	λ_{DR}	Test Acc.	Avg. Valid Acc.	RAM Usage (Mb)	Time (m)
ResNet50	10	32	Shuffle	0.	0.81	0.53	13683.84	65.08
	50	32	Shuffle	0.0002	0.84	0.55	21275.17	126.76
	50	16	Shuffle	0.0002	N/A	0.49	22183.93	264.84
	50	16	ER	0.	N/A	0.04	15017.92	22.11
	50	16	ER	0.0002	N/A	0.04	15074.38	22.37
	50	2	BER	0.0002	N/A	0.18	12088.44	220.98
ResNeSt50	50	32	Shuffle	0.00001	0.89	0.57	21571.89	154.56

[9] Jinlong Liu, Yunzhi Bai, Guoqing Jiang, Ting Chen, and Huayan Wang. Understanding why neural networks generalize well through {gsnr} of parameters. In *International Conference on Learning Representations*, 2020.

[10] Vincenzo Lomonaco and Davide Maltoni. Core50: a new dataset and benchmark for continuous object recognition. In Sergey Levine, Vincent Vanhoucke, and Ken Goldberg, editors, *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pages 17–26. PMLR, 13–15 Nov 2017.

[11] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, pages 6467–6476, 2017.

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

[13] Sebastian Mika, Gunnar Ratsch, Jason Weston, Bernhard Scholkopf, and Klaus-Robert Mullers. Fisher discriminant analysis with kernels. In *Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (cat. no. 98th8468)*, pages 41–48. Ieee, 1999.

[14] Cuong V Nguyen, Yingzhen Li, Thang D Bui, and Richard E Turner. Variational continual learning. In *International Conference on Learning Representations*, 2018.

[15] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, , and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *International Conference on Learning Representations*, 2019.

[16] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. In *Advances in Neural Information Processing Systems*, pages 348–358, 2019.

[17] 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 (IJCV)*, 115(3):211–252, 2015.

[18] Jonathan Schwarz, Jelena Luketina, Wojciech M Czarnecki, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. *arXiv preprint arXiv:1805.06370*, 2018.

[19] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. In *Advances in Neural Information Processing Systems*, pages 2990–2999, 2017.

[20] Kilian Q Weinberger, John Blitzer, and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in neural information processing systems*, pages 1473–1480, 2006.

[21] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995, 2017.

[22] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Zhi Zhang, Haibin Lin, Yue Sun, Tong He, Jonas Muller, R. Manmatha, Mu Li, and Alexander Smola. Resnest: Split-attention networks. *arXiv preprint arXiv:2004.08955*, 2020.