What is Happening Inside a Continual Learning Model? - A Representation-Based Evaluation of Representational Forgetting -

Kengo Murata
Aoyama Gakuin University
c5619156@aoyama.jp

Tetsuya Toyota
Toho University
tetsuya.toyota@is.sci.toho-u.ac.jp

Kouzou Ohara
Aoyama Gakuin University
ohara@it.aoyama.ac.jp

Abstract

Recently, many continual learning methods have been proposed, and their performance is usually evaluated based on their final output such as the class they predicted. However, this output-based evaluation cannot tell us anything about how representations the model learned from given tasks are forgotten during learning process inside the model although understanding it is important to devise a robust algorithm to catastrophic forgetting that is an intrinsic problem in continual learning. In this work, we propose a representation-based evaluation framework and demonstrate it can help us better understand the representational forgetting through intensive experiments on three benchmark datasets, which eventually brought us the following findings: 1) non-negligible amount of representational forgetting appears at shallow layers of a deep neural network model, and 2) which tasks are more accurately learned when representational forgetting occurred depends on the depth of the layer at which the representational forgetting is observed.

1. Introduction

In recent continual learning researches, many evaluation metrics [1, 2, 4, 18] and scenarios [5, 7, 15] have been proposed. In any scenario, these evaluation metrics are essentially based on the final output of a deep neural network model, *e.g.*, the class it predicted. However, such *output-based evaluations* cannot help us understand how intermediate representations of a model, *i.e.*, output vectors of its intermediate layers, are forgotten during a learning process although understanding such *representational forgetting* [12, 19] leads to developing an understanding of *catastrophic forgetting* [9, 16] that is an intrinsic problem in continual learning where the performance of a model for tasks it previously learned rapidly decreases by learning new tasks. Understanding it could enable us to devise a robust algorithm to catastrophic forgetting.

In this paper, we focus on the *representation-based evaluation* and propose an evaluation framework based on it. Besides, through intensive experiments with various continual learning methods under incremental class learning scenario [7], we demonstrate the proposed framework can help us understand the representational forgetting in more depth. Our major findings obtained from those experiments are 1) we can observe non-negligible amount of representational forgetting at shallow layers of a deep neural network model, and 2) which tasks are more accurately learned when representational forgetting occurred depends on the depth of the layer at which it occurred. We actually observed such *biases* toward the last executed task at the deepest layer and the ones toward the first executed task at shallow layers.

2. Evaluation framework

First, we provide some basic notations in this work. We refer to the output of a specific module of a deep neural network model as the *feature*. Given an input vector to the model, the corresponding output of a module is considered as a *representation* of the input. Namely, representations are instances of the feature and considered to express the knowledge learned from tasks. We say the *bias* appears on a feature if representations only for specific classes are correctly learned on the feature. Then, *representational forgetting* is defined as the phenomenon that the model forgets representations corresponding to inputs for past tasks. In other words, the bias toward the current task appears on the feature if representational forgetting occurred. This relation between representational forgetting and the bias enables us to quantitatively evaluate the strength of representational

forgetting.

Next, we introduce our representation-based evaluation framework that takes the partial retraining approach. Let F_i be the i-th feature of a deep neural network model and O_i be a set of representations that F_i outputs for input vectors of the model. Besides, let W be a set of all parameters assigned to links and units of the model and $W_{F_i} \subseteq W$ be a set of parameters used to make an output of F_i. Clearly, if representational forgetting appears on F_i, some elements in O_i are not suitable as an input of the next layer of the model and could degrade the classification accuracy. To quantitatively evaluate how the representational forgetting affects the classification performance under the incremental class learning scenario, our representation-based evaluation framework retrains a part of the model focusing on F_i as follows:

1. Train a model in continual learning setting;
2. Initialize all the parameters in $W \setminus W_{F_i}$;
3. Retrain the whole model using the data of all the tasks in offline setting with fixed parameter values in W_{F_i};
4. Iterate Steps 1 to 3 a certain times while reordering given tasks and output the average values of given evaluation metrics over all iterations.

We refer to this resulting retrained model as the partial retrained model for F_i, and express it as M_i. Our framework is an extension of the model that Xiong *et al*. used in their analysis [19]. We further added Steps 2 and 4 into their model. Initializing parameters in $W \setminus W_{F_i}$ is essential to properly evaluate the influence of the representational forgetting on the feature F_i because using values resulted from Step 1 for those parameters would bias the training at Step 3. We need Step 4 to realize the feature-oriented evaluation excluding the influence of the order of task execution. If given tasks have different complexities, the order of task execution could affect the bias on the targeted feature. To exclude such influence, we reorder the task sequence at Step 4. There are many ways of reordering the task sequence. Random shuffling is one of simplest ways. If the number of tasks is limited, circular shift that we adopted in this work is a possible systematic way of reordering. Let $T_1, T_2, \ldots, T_5$ be the initial task sequence. Then, by rotating it, we can obtain additional 4 sequences such as T_2, T_3, T_4, T_5, T_1. Averaging values in an evaluation metric over the five iterations with these sequences could exclude the influence of the complexity of a specific task from the bias analysis.

Finally, we give the definitions of our representation-based evaluation metrics, *Partial Retrain Accuracy* and F1. We define Partial Retrain Accuracy (PRA) as the overall accuracy of M_i for test data. The difference in PRA between M_i and the model trained in the offline setting that provides it with all training data of all tasks at once quantifies the strength of representational forgetting on the feature F_i. On the other hand, to clarify the direction of the bias on F_i, we define F1 of M_i as its micro-F1 with respect to a specific task. Formally, F1 of M_i for task T_j is defined as following equations. Note that, we omitted M_i from the equations for clarity.

$$\mathrm{F1}(T_j) = \frac{2\mathrm{Recall}(T_j) \cdot \mathrm{Precision}(T_j)}{\mathrm{Recall}(T_j) + \mathrm{Precision}(T_j)}, \tag{1}$$

$$\mathrm{Recall}(T_j) = \frac{\sum_{c \in C_j} \mathrm{TP}_c}{\left(\sum_{c' \in C_j} \mathrm{TP}_{c'}\right) + \left(\sum_{c' \in C_j} \mathrm{FN}_{c'}\right)}, \tag{2}$$

$$\mathrm{Precision}(T_j) = \frac{\sum_{c \in C_j} \mathrm{TP}_c}{\left(\sum_{c' \in C_j} \mathrm{TP}_{c'}\right) + \left(\sum_{c' \in C_j} \mathrm{FP}_{c'}\right)}, \tag{3}$$

where C_j is a set of classes considered in T_j and TP_c, FP_c, and FN_c are the numbers of true positives, false positives, and false negatives in the test data of class c, respectively. F1 evaluates the classification performance for a given task, so a large difference in F1 between two tasks indicates that the bias toward the task with a higher score appears on F_i.

3. Experimental setting

In our experiment, we used three datasets, MNIST [11], SVHN [17], and CIFAR10 [10]. According to the existing work [7, 20], we defined one task so that it contains two consecutive classes. As for the execution order of tasks, we adopted the rotation strategy and generated 5 task execution orders by rotating the order. We ran our framework five times with different random seeds, then we averaged their five scores to get the final results.

For MNIST, we used a CNN with three convolutional layers with ReLU activation, followed by a dense layer, and considered the output of each convolutional layer after the activation as a feature. For SVHN and CIFAR10, we used a reduced ResNet18 [14] having 6 modules and defined the output of each module as a feature. Thus, the CNN and ResNet18 respectively have 4 and 6 features including the final output of the model, which were numbered according to the depth of the corresponding layer or module. We trained these networks with Adam optimizer [8] using learning rate of 0.001 and set the number of epochs per task to 5, 10, and 10 on MNIST, SVHN, and CIFAR10, respectively. We used the same number of epochs when retraining a model using all data. In addition, at the beginning of both learning phases of our evaluation framework, we initialized the parameters of convolutional layers by He initialization [6] and those of dense layers by the normal distribution with the standard deviation 0.01. We conducted all experiments with single-headed setup [5].

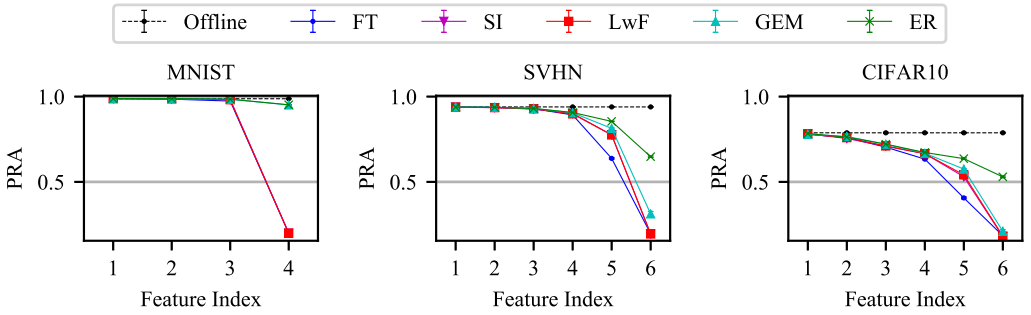

Figure 1. Scores and standard deviations of PRA for each partial retrained model on the datasets (left: MNIST, middle: SVHN, right: CIFAR10). Error bars of standard deviations are not visible due to too small values.

With the above experimental settings, we compared four continual learning methods: Synaptic Intelligence (SI) [20], Learning without Forgetting (LwF) [13], Gradient Episodic Memory [14], and Experience Replay (ER) [3] that is also called as Naive Rehearsal [7]. Furthermore, we adopted Fine-tuning (FT) and Offline. FT trains a model on all tasks sequentially in the standard way, so it gives a lower bound in a performance metric. On the other hand, Offline trains a model using the data of all tasks in offline setting and gives the upper bound in the metric. As for the hyper parameters of these methods, we set the regularization coefficient in SI and LwF to 1.0, and the number of stored samples per class in GEM and ER to 128, 128, and 512 for MNIST, SVHN and CIFAR10, respectively. Note that, we applied random sampling strategy for ER as with [7], while we took the FIFO strategy for GEM according to its original paper [14].

4. Experimental results

We firstly investigate the resulting scores in PRA shown in Figure 1, where the scores at the largest feature index correspond to the accuracy of the model resulted from applying continual learning to it. Note that the results for Offline that does not involve retraining are a fixed value regardless of feature indices on every dataset. Comparing the other results with them, we can know at which feature representational forgetting occurs and how strong it is. In fact, on each dataset, it is observed that the strength of representational forgetting is getting larger as the depth of feature becomes deeper. Here, it is worth noting that, on MNIST, we can observe the difference between Offline and the other methods only at the 4th feature corresponding to the output layer of the model. This implies that, in this case, catastrophic forgetting occurs, but representational forgetting does not. Thus, we exclude the results on MNIST from the succeeding investigation. By contrast, we can find representational forgetting occurs on SVHN and CIFAR10, but the results on them exhibit different tendencies. The smallest feature index at which we can find a negligible difference between

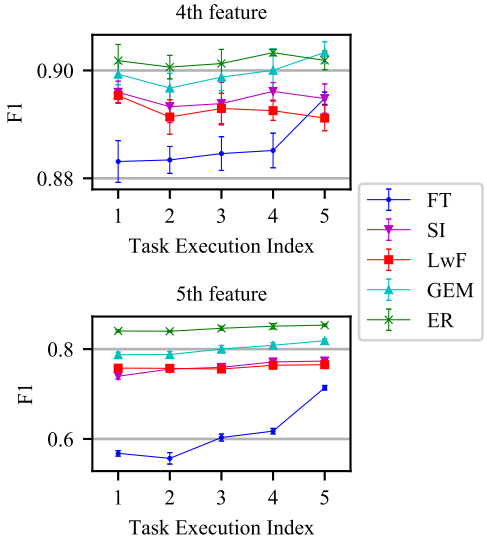

Figure 2. Scores and standard deviations of F1 for each task on SVHN (upper: 4th feature, lower: 5th feature).

Offline and the other methods is 2 on CIFAR10, while 4 on SVHN. Hereafter, we refer to the feature having this smallest index as the starting feature. Considering the complexity of the task, these results suggest that representational forgetting occurs at a shallower feature as given tasks become more complex. The baseline upper-bound score given by Offline for CIFAR10 is smaller than that for SVHN, which means the classification tasks in CIFAR10 are more complex and difficult than those in SVHN.

Next, we consider the differences between the learning methods in terms of the strength of representational forgetting. In Figure 1, it can be seen that the strength of representational forgetting depends on both the dataset and the learning method unlike the starting feature that depends only on the dataset. More specifically, both on SVHN and CIFAR10, there exists a large difference between ER and

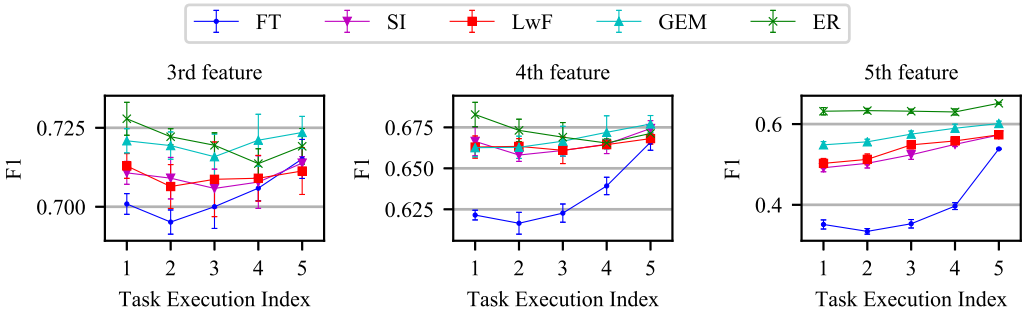

Figure 3. Scores and standard deviations of F1 for each task on CIFAR10 (left: 3rd feature, middle: 4th feature, right: 5th feature).

FT in the strength of representational forgetting. Comparing with FT, it is found that also SI, LwF, and GEM succeed in alleviating the representational forgetting at the 5th feature. However, unfortunately, SI and LwF cannot reduce the catastrophic forgetting at the output layer. These results suggest that the regularization based methods such as SI and LwF can alleviate representational forgetting at a deep feature, but they are less effective on catastrophic forgetting, while the sample reuse based methods such as GEM and ER are effective for alleviating both representational forgetting at a deep layer and catastrophic forgetting. Especially, ER is highly effective for reducing catastrophic forgetting. In contrary, there is little difference between the methods in the strength of representational forgetting at a feature that is closer to the starting one, say the 2nd or the 3rd feature on CIFAR10. This observation indicates that the advantage of the sophisticated continual learning methods over the other ones is limited to the features close to the output layer.

We next show the scores of F1 for each dataset for the features on which representational forgetting that has strength of a certain degree is observed. Figure 2 shows the scores of F1 for the 4th and 5th features on SVHN. For the 4th feature, only FT shows the strong bias toward the 5th executed task. On the other hand, most of the methods exhibit an increasing tendency for the 5th feature, which implies that there exist the biases toward the later tasks on the 5th feature. Figure 3 shows the scores of F1 on the 3rd, 4th, and 5th features on CIFAR10. We first found that all the methods exhibit the monotonically increasing tendency for the 5th feature as with the case of SVHN. In contrast, all the methods except for FT display the valley-like shape for the 3rd feature, which implies the existence of the biases toward both the 1st and 5th executed tasks. In particular, ER takes a much greater value for the 1st executed task than the value for the 5th executed task, so we can say ER has a strong bias toward the 1st executed task on the 3rd feature. As for the 4th feature, the direction of the bias is different according to the method. GEM only shows the bias toward the new tasks, on the other hand, SI and ER still show the biases toward the 1st and 5th executed tasks. Note that FT shows the biases toward the later tasks on all the features.

In summary, these results suggest that the biases toward the later tasks are getting stronger as the depth of the feature becomes deeper, while the biases toward the 1st executed task appears on other features for the learning methods that have a mechanism to alleviate catastrophic forgetting. Note that the biases toward the former tasks never appear on all the features in the case of FT that has no such mechanism. This finding indicates that these models tend to forget representations for the past tasks on the features at deep layers, while keeping some useful representations for the past tasks, especially for the 1st executed one, on the features at the middle layers. It is worth noting that, although the ability to mitigate catastrophic forgetting could allow us to preserve representations for past tasks at middle layers of a model, at the same time, it prevents the model from learning new tasks at those layers. This is justified by the fact that the scores of F1 for the later tasks are relatively low on the features at the middle layer. In other words, these continual learning methods cannot consolidate representations for new tasks into ones for past tasks well at the middle layers.

5. Conclusion

In this paper, we proposed a representation-based evaluation framework and demonstrated it can help us understand representational forgetting in more depth. We believe that the findings in this study could provide a new insight into catastrophic forgetting and contribute to developing more robust algorithms to catastrophic forgetting. The datasets used in our experiments only have 5 binary classification tasks. Thus, to generalize our findings, we are planning to conduct further experiments using longer task sequences as one of immediate future work. Besides, based on the observations in this work, we will also devise a continual learning algorithm that can merge representations for a new task into ones for past tasks without degrading F1 scores at middle layers.

References

[1] Massimo Caccia, Pau Rodriguez, Oleksiy Ostapenko, Fabrice Normandin, Min Lin, Lucas Caccia, Issam Laradji, Irina Rish, Alexande Lacoste, David Vazquez, et al. Online fast adaptation and knowledge accumulation: a new approach to continual learning. *arXiv preprint arXiv:2003.05856*, 2020.

[2] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *ICLR*, 2019.

[3] 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.

[4] Natalia Díaz-Rodríguez, Vincenzo Lomonaco, David Filliat, and Davide Maltoni. Don't forget, there is more than forgetting: new metrics for continual learning. In *NeurIPS Workshop*, 2018.

[5] Sebastian Farquhar and Yarin Gal. Towards robust evaluations of continual learning. *arXiv preprint arXiv:1805.09733*, 2018.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, pages 1026–1034, 2015.

[7] Yen-Chang Hsu, Yen-Cheng Liu, Anita Ramasamy, and Zsolt Kira. Re-evaluating continual learning scenarios: A categorization and case for strong baselines. *arXiv preprint arXiv:1810.12488*, 2018.

[8] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[9] 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. *PNAS*, 114(13):3521–3526, 2017.

[10] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[11] Yann LeCun, Corinna Cortes, and Christopher JC Burges. The mnist database of handwritten digits, 1998. *URL http://yann.lecun.com/exdb/mnist*.

[12] Timothée Lesort, Hugo Caselles-Dupré, Michael Garcia-Ortiz, Andrei Stoian, and David Filliat. Generative models from the perspective of continual learning. In *IJCNN*, pages 1–8, 2019.

[13] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE TPAMI*, 40(12):2935–2947, 2017.

[14] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *NeurIPS*, pages 6467–6476, 2017.

[15] Davide Maltoni and Vincenzo Lomonaco. Continuous learning in single-incremental-task scenarios. *Neural Networks*, 116:56–73, 2019.

[16] 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. 1989.

[17] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NeurIPS Workshop*, 2011.

[18] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *ICML*, volume 80, pages 4548–4557, 2018.

[19] Yuwen Xiong, Mengye Ren, and Raquel Urtasun. Learning to remember from a multi-task teacher. *arXiv preprint arXiv:1910.04650*, 2019.

[20] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *ICML*, volume 70, pages 3987–3995, 2017.