EWC for CVPR 2020 Workshop Challenge on Continual Learning in Computer Vision

Haoran Zhu
New York University
hz1922@nyu.edu

Abstract

Continual learning is a crucial ability for artificial intelligence. It aims at allowing an agent to learn new tasks without forgetting the knowledge of previous tasks. In real life applications(e.g., autonomous driving cars, robotic applications), due to limited computational and storage resources, it is almost impossible to fully retrain models every time new tasks and new data become available. In this work, we take part in CVPR 2020 Workshop on Continual Learning in Computer Vision Challenge. We apply EWC[1] algorithm on a real life dataset, CORe50, which is a new dataset and benchmark specially designed for continuous object recognition scenarios. In New Classes task, EWC algorithm ranks at the 7th place among all participants in the world.

1. Introduction

In today's Deep Learning scheme, the presumption of the training is that all instances of the classes are available, the model will iterate the dataset per epoch to learn the knowledge. This means if there are new instances added to the dataset, we have to retrain the whole model on the whole dataset plus the new instances. This has caused trouble in some scenarios. Specifically, in image classification problems, the pretrained model often encounters new objects or the dataset can be expanded. However, the existence of Catastrophic Forgetting[2], i.e. the newly learned parameters will shift the old parameters and weaken its performance on old task, has brought challenge to this problem. This phenomenon can greatly degrade the performance of the model or even totally rewrite the model's parameters, causing the old knowledge being totally forgot. In before, in face of such case, one have to choose either retrain the model on entire dataset or just tolerate such degrading issues. To overcome the dilemma, the concept of continual learning(Lifelong Learning[3]) has emerged. In continual learning, it is required that the model must have not only the ability to acquire new knowledge, but also prevent the novel input to overwhelm the original data.

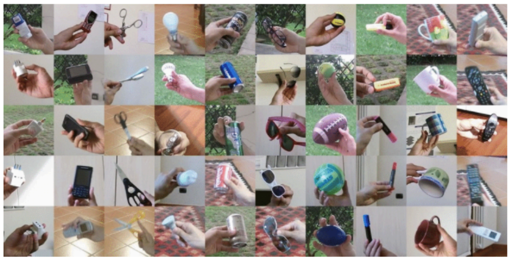


Figure 1. Example images of the 50 objects in CORe50. Each column denotes one of the 10 categories

2. Dataset and Tasks

CORe50 is a new dataset specially designed for (C)ontinual (O)bject (Re)cognition. Unlike permuted MNIST where new tasks to learn are obtained by simply scrambling the pixel positions, CORe50 is much more complex and a real life dataset. Datasets such as ImageNet and Pascal VOC provide a good playground for image classification and detection, but they have been designed with "static" evaluation and lack of multiple views of the same objects taken into different sessions, CORe50 solves the above problems and meets the requirement for continuous learning scenarios on computer vision.

It consists 50 domestic objects belonging to 10 categories. The dataset is separated into 11 distinct sessions (8 indoors and 3 outdoors) with different background and lightning. For each session and for each object, a 15 seconds video (at 20 fps) has been recorded with a Kinect 2.0 sensor delivering 300 RGB-D frames.

There are three tasks in CVPR 2020 Workshop CLVision Challenge:

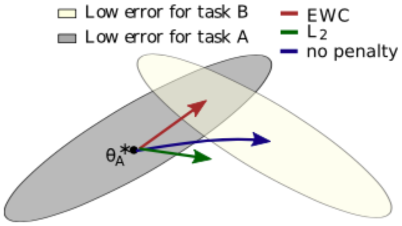


Figure 2. By applying EWC, the model of the weights can learn the new task while still maintain in the range of solving the old task. (Red arrow)

- New Classes(NC): New training patterns belonging to different classes become available in subsequent batches. In this case the model should be able to deal with the new classes without losing accuracy on the previous ones.
- New Instances(NI): New training patterns of the same classes become available in subse- quent batches with new poses and conditions (illumination, background, occlusion, etc.). A good model is expected to incrementally consolidate its knowledge about the known classes without compromising what it has learned before.
- New Instances and Classes(NIC): New training patterns belonging both to known and new classes become available in subsequent training batches. A good model is expected to consolidate its knowledge about the known classes and to learn the new ones.

For each task, the input image is $128 \times 128 \times 3$. For case NC and NI, each task batch contains 10k-20k images, which makes it a large dataset.

3. Introduction

In brains, synaptic consolidation enables continual learning by reducing the plasticity of synapses that are vital to previously learned tasks. EWC is an algorithm that performs a similar operation in artificial neural networks by constraining important parameters to stay close to their old values. EWC can be used in supervised learning and reinforcement learning problems to train several tasks sequentially without forgetting older ones.

3.1. Theorem

From a probabilistic perspective: optimizing the parameters is tantamount to finding their most probable values given some data D. We can compute this conditional probability $p(\theta|\mathcal{D})$ from the prior probability of the parameters $p(\theta)$ and the probability of the data $p(\mathcal{D}|\theta)$ by using Bayes' rule:

$$\log p(\theta|\mathcal{D}) = \log p(\mathcal{D}|\theta) + \log p(\theta) - \log p(\mathcal{D}) \tag{1}$$

Similarly we can apply Bayes' rule on two continuous tasks A and B:

$$\log p(\theta|\mathcal{D}) = \log p\left(\mathcal{D}_B|\theta\right) + \log p\left(\theta|\mathcal{D}_A\right) - \log p\left(\mathcal{D}_B\right) \tag{2}$$

While $\log p(\mathcal{D}_B|\theta)$ is negative training loss for task B, we can optimize this by gradient descent. $\log p(\mathcal{D}_B)$ is a constant, we don't need to care about it when optimizing for the overall goal. $\log p(\theta|\mathcal{D}_A)$'s real value is intractable and will be approximated by Laplace approximation. Suppose parameter θ is a Gaussian distribution with average $\hat{\theta}$. The approximated value can further be the derivation regarding approximated mean value and its corresponding Hessian matrix:

$$\begin{aligned} h(\theta|\mathcal{D}_A) &= \log p(\theta|\mathcal{D}_A) \\ &\approx h(\hat{\theta}|\mathcal{D}_A) + (\theta-\hat{\theta})h'(\hat{\theta}|\mathcal{D}_A) + \frac{1}{2}(\theta-\hat{\theta})^2 h''(\hat{\theta}|\mathcal{D}_A) \\ &= \text{const} + \frac{1}{2}(\theta-\hat{\theta})^2 h''(\theta|\mathcal{D}_A) \end{aligned}$$

Noted that the expected value of Hessian matrix is the negative value of Fisher information matrix F, thus the original goal of maximizing $\log p(\theta|\mathcal{D})$ is equivalent to minimizing the following loss function:

$$\mathcal{L}(\theta) = \mathcal{L}_B(\theta) + \sum_i \frac{\lambda}{2} F_i \left(\theta_i - \theta^*_{A,i}\right)^2 \tag{3}$$

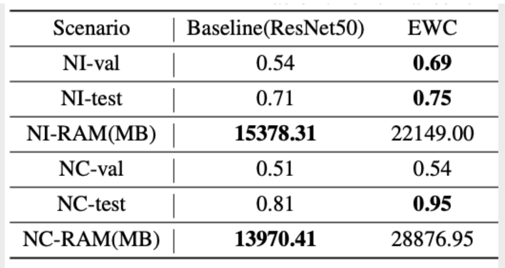
where $\mathcal{L_B}(\theta)$ is the loss for task B only and hyperparameter λ sets how important the old task is compared to the new task. When the third task comes, you can treat all previous tasks as the old task and repeat the above process. In the end, it can ensure the model can learn new tasks without changing the weights for the previous tasks too much.

3.2. Experiment Setting

After several experiments, we set the hyperparameter $\lambda = 100$, which we think it is an appropriate value of how important the old task is to the new task. The best results are produced by SGD as the optimizer.

4. Evaluation and Analyze

In this part, we evaluate the performance of EWC. We will discuss the performance for them on NI(New



Scenario	Baseline(ResNet50)	EWC
NI-val	0.54	**0.69**
NI-test	0.71	**0.75**
NI-RAM(MB)	**15378.31**	22149.00
NC-val	0.51	0.54
NC-test	0.81	**0.95**
NC-RAM(MB)	**13970.41**	28876.95

Figure 3. Performance of EWC compared to baseline model

Instance) case and NIC(New Instance and Class) case respectively. Then we will analyze their performance case by case. According to the guideline of the CLVision Challenge, we use average accuracy, i.e. the mean accuracy per task as the metrics. Besides, metrics including maximum RAM usage, average RAM usage and training time should also be taken into account to validate the performance.

4.1. EWC

In this part, due to the requirements of GPU memory, all of our experiments are done with a linux server with Intel Broadwell CPU platform, 52 GB RAM and Tesla T4 GPU with 16GB SDRAM. In all our cases, we set batch size to 32, with learning rate at 0.01. To achieve better performance, we will load the pre-trained ResNet50 and apply EWC on all following tasks.

The summarization of the performance result is shown in table. Here *-val represents the performance on validation set, *-test is the performance on test data set. Accuracy on validation dataset is the average of per-task accuracy. Throughout the training process, the validationset is fixed. Therefore, this index can represent how much the model have mitigated the catastrophic forgetting.

- NI From the perspective of validatoin accuracy, EWC performs the best, scoring 0.69 in validation and 0.75 in test. It demonstrates EWC's ablility to reach and maintain high accuracy after iterating through 8 batches of new instance data. However, due to EWC have to retrain the whole network when encountering new task, it consumes much more RAM than the baseline.

- NC In NC case, EWC is still the best in test accuracy. It ranks at the 7th place among all participants in the world. EWC and baseline are undertraining in the first several batches, causing the low performance at the beginning. However, EWC ends up with a high accuracy after it sees more and more samples, EWC still uses the most RAM.

5. Conclusion

In this project, we have looked thoroughly into the literature and legacy work of continual learning and then we choose EWC to experiment on the CLVision Challenge issued by CVPR 2020 Workshop. For EWC, the performance is hugely improved compared to baseline model. For New Classes task, we rank at the 7th place among all participants in the world.

References

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

[2] 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.

[3] Sebastian Thrun. A lifelong learning perspective for mobile robot control. In Intelligent Robots and Systems, pages 201–214. Elsevier, 1995.