2022 / 09 / 02

Introduction to Continual/Incremental Learning

Seokho Moon

danny232@korea.ac.kr

School of Industrial and Management Engineering, Korea University

❖ 문석호 (Seokho Moon)

- 고려대학교 산업경영공학부 졸업(2019.08)
- 고려대학교 산업경영공학과 대학원 재학 중
- Data Mining & Quality Analytics Lab (김성범 교수님)
- 석박사통합과정 (2019.09 ~)

❖ 관심 연구 분야

- Continual/Incremental Learning
- Self/Semi-supervised learning
- Anomaly Detection

❖ E-mail

- danny232@korea.ac.kr

Reference

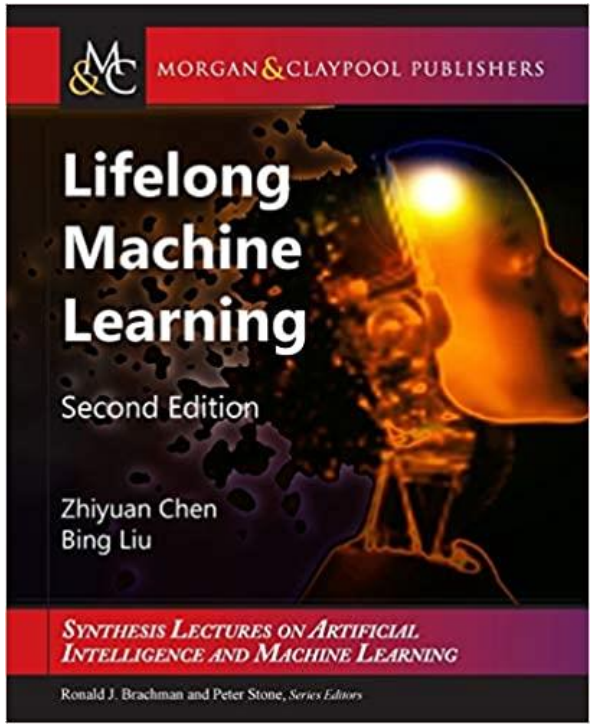
- 본 세미나 자료를 만들면서 많이 참고한 자료이며, 더욱 자세히 설명되어 있음

Lifelong and Continual Learning

Part I – Slides for June 14, 2022

Bing Liu and **Zixuan Ke**
Department of Computer Science
University of Illinois at Chicago

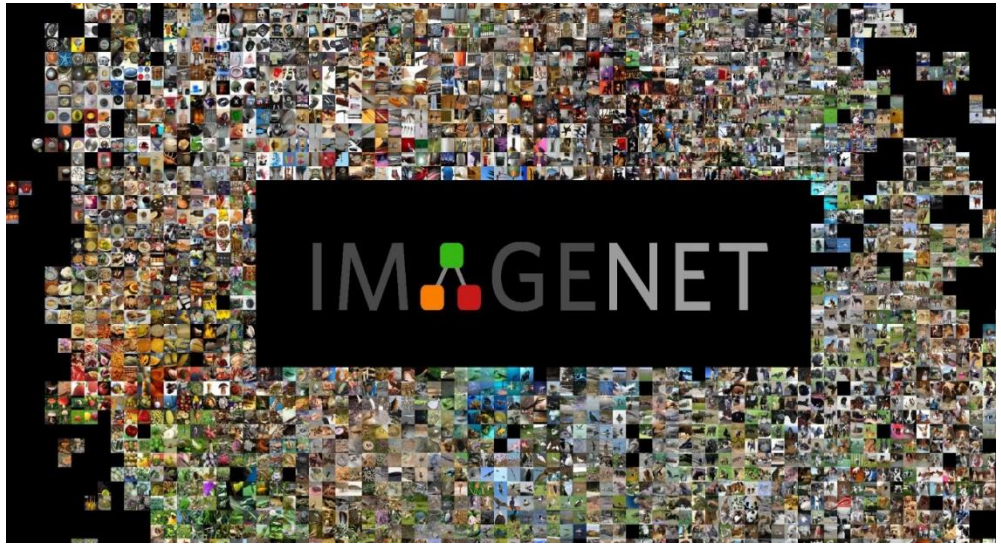
A short PhD course (8 hours) given at Aalborg University on June 14 and June 16, 2022

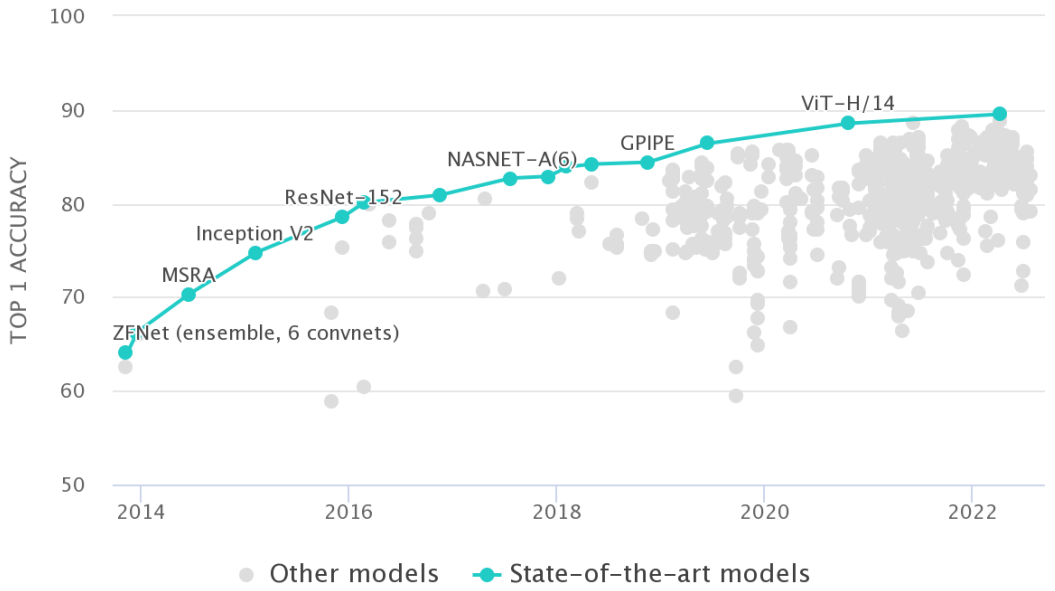


https://www.cs.uic.edu/~liub/Part-1-continual-learning-slides.pdf
https://www.cs.uic.edu/~liub/lifelong-machine-learning.html

Introduction

- 현재의 machine learning 및 deep learning은 특정 task에서 매우 우수한 성능을 보여주고 있음
- 예를 들어 image classification의 경우, 인간의 분류 성능을 이미 넘어섰으며 지속적으로 성능이 증가하는 상황





https://paperswithcode.com/sota/image-classification-on-imagenet

Introduction

- 그러나 앞선 예시의 ImageNet에서 학습하지 않은 데이터에 대한 inference 성능이 저하되는 등의 문제가 있으며, 인간과 유사한 Artificial Intelligence (AI) 으로 보기에는 **여러가지 한계점**이 존재

1. 지속적으로 학습하지 않으며, knowledge가 **축적/이동되지 않음**
2. 문제 상황이 **close-world**를 가정하고 있어서, 데이터가 새롭게 추가되는 실제 현실에 대한 대응이 어려움
3. 한번 학습이 완료된 모델을 test task에 적용한 후에, 해당 모델이 **새롭게 학습되지 않음**

Liu, B., & Mazumder, S. (2021, May). Lifelong and continual learning dialogue systems: learning during conversation. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 35, No. 17, pp. 15058-15063).

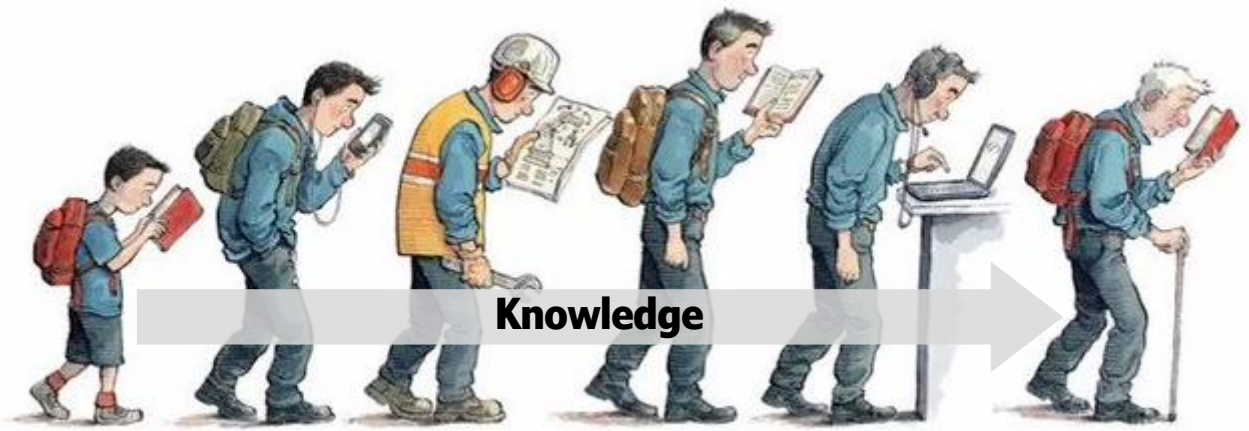
Introduction

- 그러나 앞선 예시의 ImageNet에서 학습하지 않은 데이터에 대한 inference 성능이 저하되는 등의 문제가 있으며, 인간과 유사한 Artificial Intelligence (AI) 으로 보기에는 **여러가지 한계점**이 존재

1. 지속적으로 학습하지 않으며, knowledge가 **축적/이동되지 않음**
2. 문제 상황이 **close-world**를 가정하고 있어서, 데이터가 새롭게 추가되는 실제 현실에 대한 대응이 어려움
3. 한번 학습이 완료된 모델을 test task에 적용한 후에, 해당 모델이 **새롭게 학습되지 않음**

이러한 문제를 해결하기 위해
continual learning 방식을 제안

Continual Learning of Humans



https://www.economist.com/special-report/2017/01/12/lifelong-learning-is-becoming-an-economic-imperative

Liu, B., & Mazumder, S. (2021, May). Lifelong and continual learning dialogue systems: learning during conversation. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 35, No. 17, pp. 15058-15063).

What is continual learning?

- Continual learning은 training data에 접근할 수 없는(가능하더라도 거의 적은) 현실적인 상황에서도, 하나의 neural network(and limited resource)로 여러 개의 sequential tasks의 **성능을 유지**하거나 **향상**시키는 것
- 특징 :

1. 말 그대로 data stream이 지속적인 형태에서의 학습 방식을 의미
2. Knowledge base(KB)에 knowledge를 축적
3. Knowledge를 task를 교차하여 transferring 진행
   - Forward transfer: 이전 tasks의 knowledge를 통해 미래의 task 성능을 향상
   - Backward transfer: 미래 task의 knowledge를 사용하여 이전 tasks의 성능을 향상

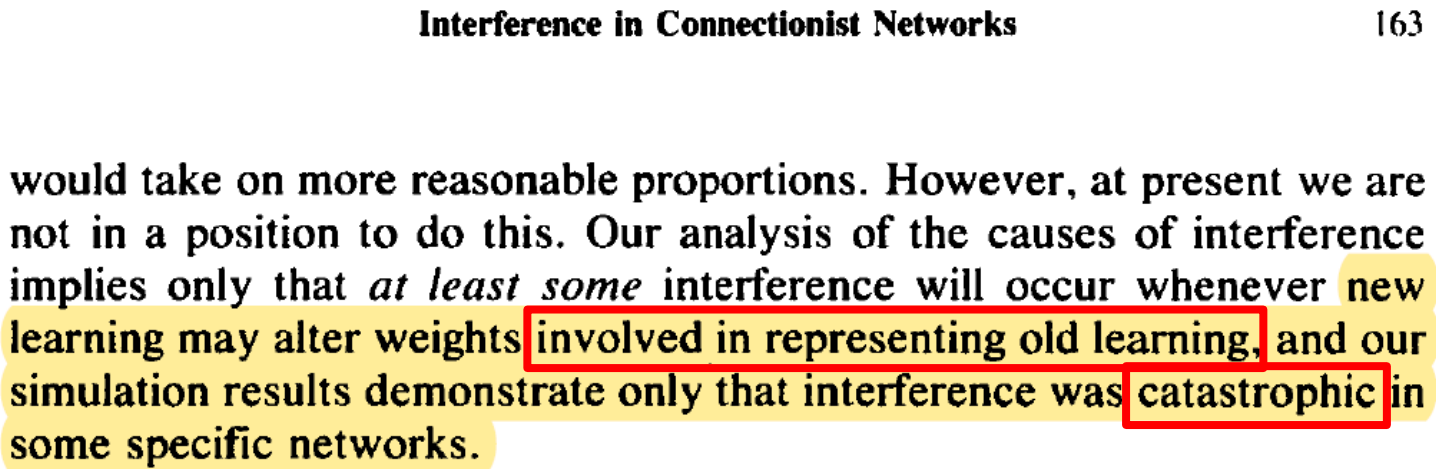
1. 지속적으로 학습하지 않으며, knowledge가 **축적/이동되지 않음**
2. 문제 상황이 **close-world**를 가정하고 있어서, 데이터가 새롭게 추가되는 실제 현실에 대한 대응이 어려움
3. 한번 학습이 완료된 모델을 test task에 적용한 후에, 해당 모델이 **새롭게 학습되지 않음**

To address the limitations,
computers should adopt continual learning method as humans.

https://www.reddit.com/r/2D3DAI/comments/js69za/references_from_lecture_introduction_to_continual/

Catastrophic forgetting

- 하나의 neural network로 여러 개의 sequential tasks를 순차적으로 학습시키는 경우, 새로운 task를 학습시키면 이전 tasks의 정보를 잃어서 성능이 저하되는 **catastrophic forgetting**이 발생
  - Gradient-based 형태의 학습 구조에서 종종 나타나는 문제라고 McCloskey and Cohen가 1989년 최초로 언급
  - 인간의 학습방식처럼 기존 지식이 없어지지 않도록 많은 연구들이 제안되고 있음



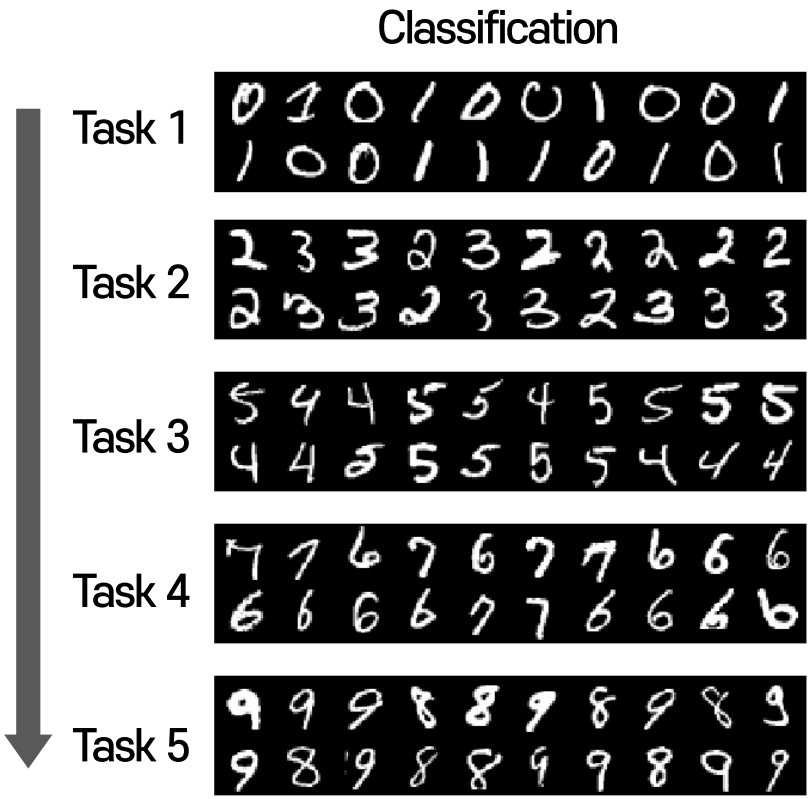
Interference in Connectionist Networks 163

would take on more reasonable proportions. However, at present we are not in a position to do this. Our analysis of the causes of interference implies only that *at least some* interference will occur whenever new learning may alter weights involved in representing old learning, and our simulation results demonstrate only that interference was catastrophic in some specific networks.

McCloskey, M., & Cohen, N. J. (1989). Catastrophic interference in connectionist networks: The sequential learning problem. In Psychology of learning and motivation (Vol. 24, pp. 109-165). Academic Press.

Example of Tasks

- MNIST 데이터셋을 예시로 사용하여 2 class classification을 하는 5개의 task를 도식화
- 각 task는 1→2→3→4→5 의 순서를 가지고 **순차적**으로 학습이 진행되며, network는 1개로 구성



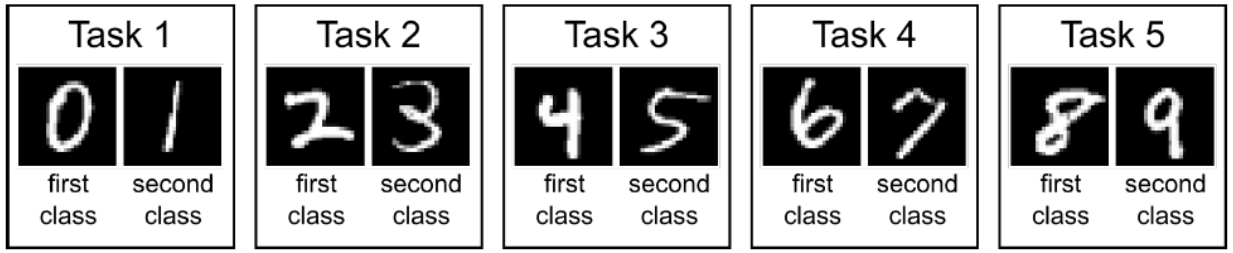


Figure 1: Schematic of split MNIST task protocol.

LeCun, Yann. "The MNIST database of handwritten digits."http://yann. lecun. com/exdb/mnist/(1998).
Van de Ven, G. M., & Tolias, A. S. “Three scenarios for continual learning.” (2018). Neural Information Processing Systems Workshops, 2018.

Three continual learning scenarios

- Continual learning은 크게 3가지의 scenario를 가지며, 각각은 task incremental learning(**Task-IL**), domain incremental learning(**Domain-IL**), and class incremental learning(**Class-IL**) 으로 되어 있음

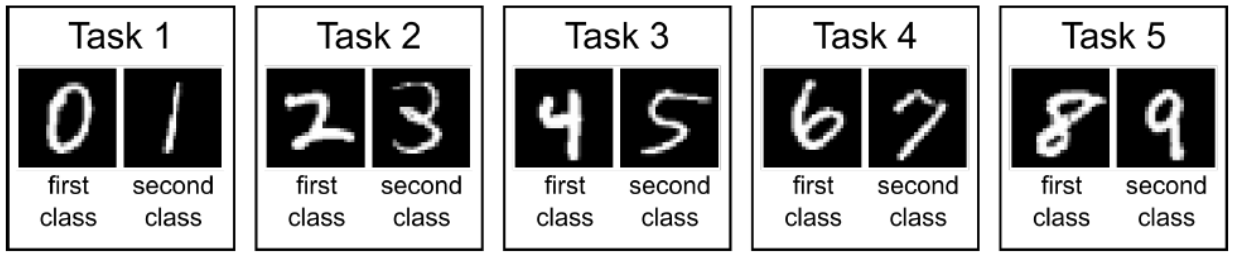


Figure 1: Schematic of split MNIST task protocol.

Table 1: Overview of the three continual learning scenarios.

Scenario	*Required at test time*
Task-IL	Solve tasks so far, task-ID provided
Domain-IL	Solve tasks so far, task-ID not provided
Class-IL	Solve tasks so far *and* infer task-ID

Table 2: Split MNIST according to each scenario.

Task-IL	With task given, is it the 1st or 2nd class? (e.g., 0 or 1)
Domain-IL	With task unknown, is it a 1st or 2nd class? (e.g., in [0, 2, 4, 6, 8] or in [1, 3, 5, 7, 9])
Class-IL	With task unknown, which digit is it? (i.e., choice from 0 to 9)

LeCun, Yann. "The MNIST database of handwritten digits."http://yann. lecun. com/exdb/mnist/(1998).
Van de Ven, G. M., & Tolias, A. S. “Three scenarios for continual learning.” (2018). Neural Information Processing Systems Workshops, 2018.

Three continual learning scenarios

- Task 5까지 순차적으로 학습이 완료된 1개의 network를 사용하는 상황 (test 시 목표에 따라 구분)

1. **Task-IL :** 특정 task(e.g., task 4)를 선정하고 학습된 network를 통해 해당 task의 성능(e.g., accuracy)을 확인
2. **Domain-IL :** 각 Task에 사용된 데이터의 domain이 다른 상황에서, 학습된 network의 task 성능을 확인(task는 동일)
3. **Class-IL :** 모든 task에 사용된 class를 전부 통합하여, 기존에 학습된 network를 통해 class 분류 성능 확인

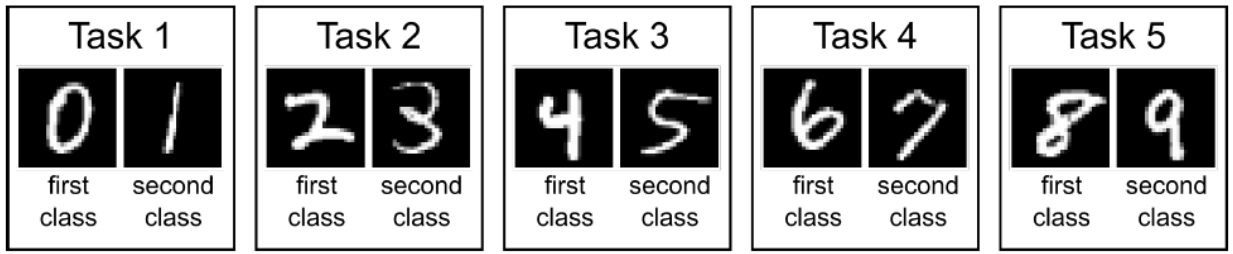


Figure 1: Schematic of split MNIST task protocol.

Table 1: Overview of the three continual learning scenarios.

Scenario	*Required at test time*
Task-IL	Solve tasks so far, task-ID provided
Domain-IL	Solve tasks so far, task-ID not provided
Class-IL	Solve tasks so far *and* infer task-ID

Table 2: Split MNIST according to each scenario.

Task-IL	With task given, is it the 1^{st} or 2^{nd} class? (e.g., 0 or 1)
Domain-IL	With task unknown, is it a 1^{st} or 2^{nd} class? (e.g., in [0, 2, 4, 6, 8] or in [1, 3, 5, 7, 9])
Class-IL	With task unknown, which digit is it? (i.e., choice from 0 to 9)

LeCun, Yann. "The MNIST database of handwritten digits."http://yann. lecun. com/exdb/mnist/(1998).
Van de Ven, G. M., & Tolias, A. S. "Three scenarios for continual learning." (2018). Neural Information Processing Systems Workshops, 2018.

Three continual learning scenarios

- Task 5까지 순차적으로 학습이 완료된 1개의 network를 사용하는 상황 (test 시 목표에 따라 구분)

1. **Task-IL :** 특정 task(e.g., task 4)를 선정하고 학습된 network를 통해 해당 task의 성능(e.g., accuracy)을 확인
2. **Domain-IL :** 각 Task에 사용된 데이터의 domain이 다른 상황에서, 학습된 network의 task 성능을 확인(task는 동일)
3. **Class-IL :** 모든 task에 사용된 class를 전부 통합하여, 기존에 학습된 network를 통해 class 분류 성능 확인

Domain-IL example

Domain 1 — Task

One task is to classify car reviews as positive or negative and another task is to classify camera reviews as positive or negative. Car and camera are two domains.

Domain 2 — Task

Class-IL example

Domain : animal — Task 1

Today we learn to recognize pig and chicken,

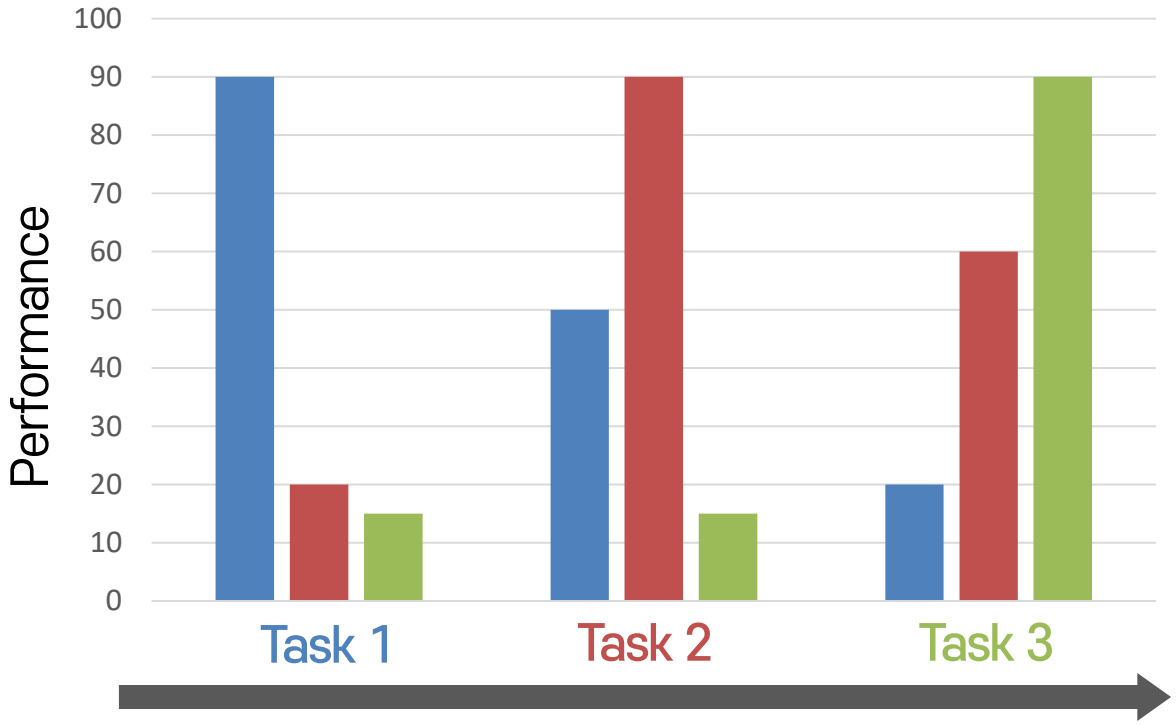
and tomorrow, we also learn to recognize sheep.

Task 2

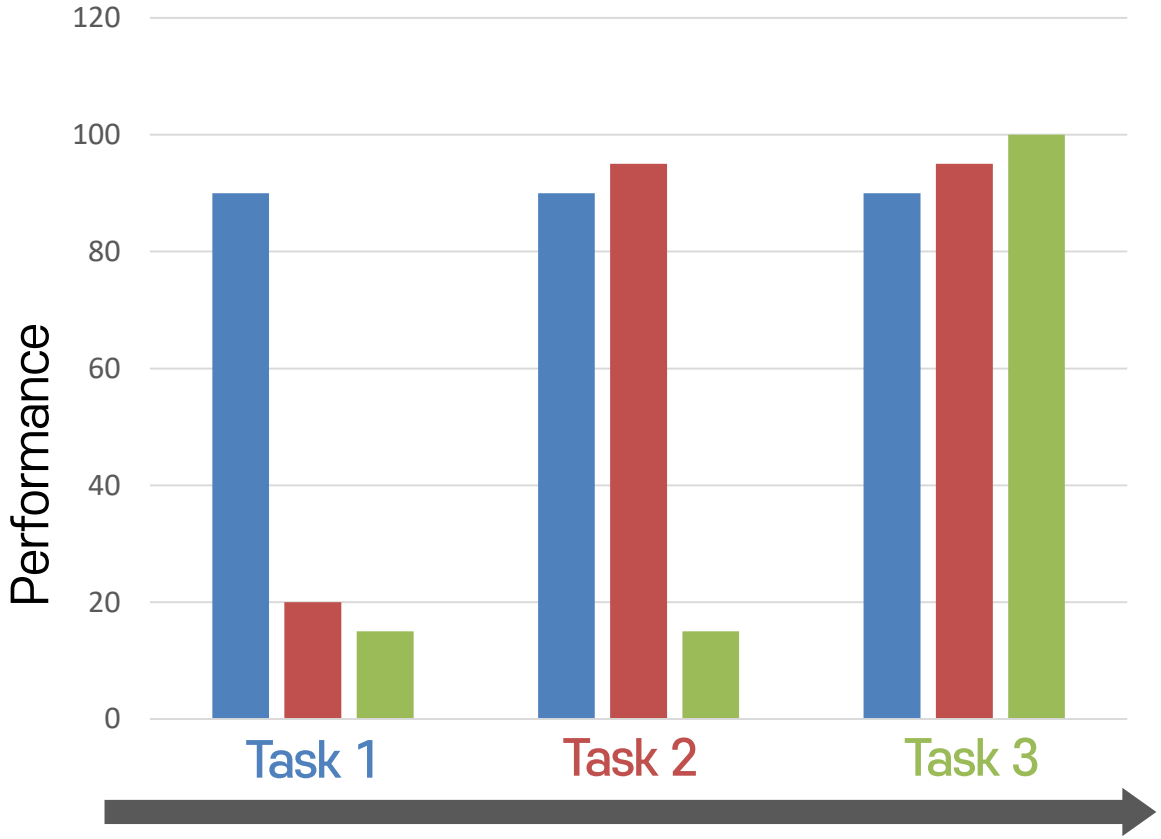
Van de Ven, G. M., & Tolias, A. S. "Three scenarios for continual learning." (2018). Neural Information Processing Systems Workshops, 2018.

Task-IL scenario

- Task-IL scenario에서 catastrophic forgetting과 (knowledge) forward transfer 상황 설명



Scenario : Catastrophic forgetting



Scenario : Forward transfer

Approaches

- Task-IL, Domain-IL, Class-IL 실험 성능 비교 예시 (2018년 기준)
- Split MNIST 문제 상황에서는 난이도가 Task-IL → Domain-IL → Class-IL 순으로 증가함을 알 수 있음
- Baseline에서 **offline**은 한번에 모든 training data에 접근하여 학습한 **multitask-learning**을 의미

Table 4: Average test accuracy (over all tasks) on the split MNIST task protocol. Each experiment was performed 20 times with different random seeds, reported is the mean (± SEM) over these runs.

Approach	Method	Task-IL	Domain-IL	Class-IL
Baselines	*None – lower bound*	*87.19 (± 0.94)*	*59.21 (± 2.04)*	*19.90 (± 0.02)*
	Offline – upper bound	*99.66 (± 0.02)*	*98.42 (± 0.06)*	*97.94 (± 0.03)*
Task-specific	XdG	99.10 (± 0.08)	-	-
Regularization	EWC	98.64 (± 0.22)	63.95 (± 1.90)	20.01 (± 0.06)
	Online EWC	99.12 (± 0.11)	64.32 (± 1.90)	19.96 (± 0.07)
	SI	99.09 (± 0.15)	65.36 (± 1.57)	19.99 (± 0.06)
Replay	LwF	99.57 (± 0.02)	71.50 (± 1.63)	23.85 (± 0.44)
	DGR	99.50 (± 0.03)	95.72 (± 0.25)	90.79 (± 0.41)
	DGR+distill	99.61 (± 0.02)	96.83 (± 0.20)	91.79 (± 0.32)
Replay + Exemplars	iCaRL (budget = 2000)	-	-	94.57 (± 0.11)

Van de Ven, G. M., & Tolias, A. S. "Three scenarios for continual learning." (2018). Neural Information Processing Systems Workshops, 2018.

Approaches

- **Approaches :** 1) Regularization-based; 2)Replay-based; 3)Architecture-based

1) **Regularization-based :** loss function에 regularization term을 추가해서 forgetting 을 방지

2) **Replay-based :** memory를 통해서 training subset을 만들어서 학습하거나, data generator 를 학습

3) **Architecture-based :** task에 대해 각각 sub-network 를 가지거나, network가 dynamic expandable하는 형태

Table 4: Average test accuracy (over all tasks) on the split MNIST task protocol. Each experiment was performed 20 times with different random seeds, reported is the mean (± SEM) over these runs.

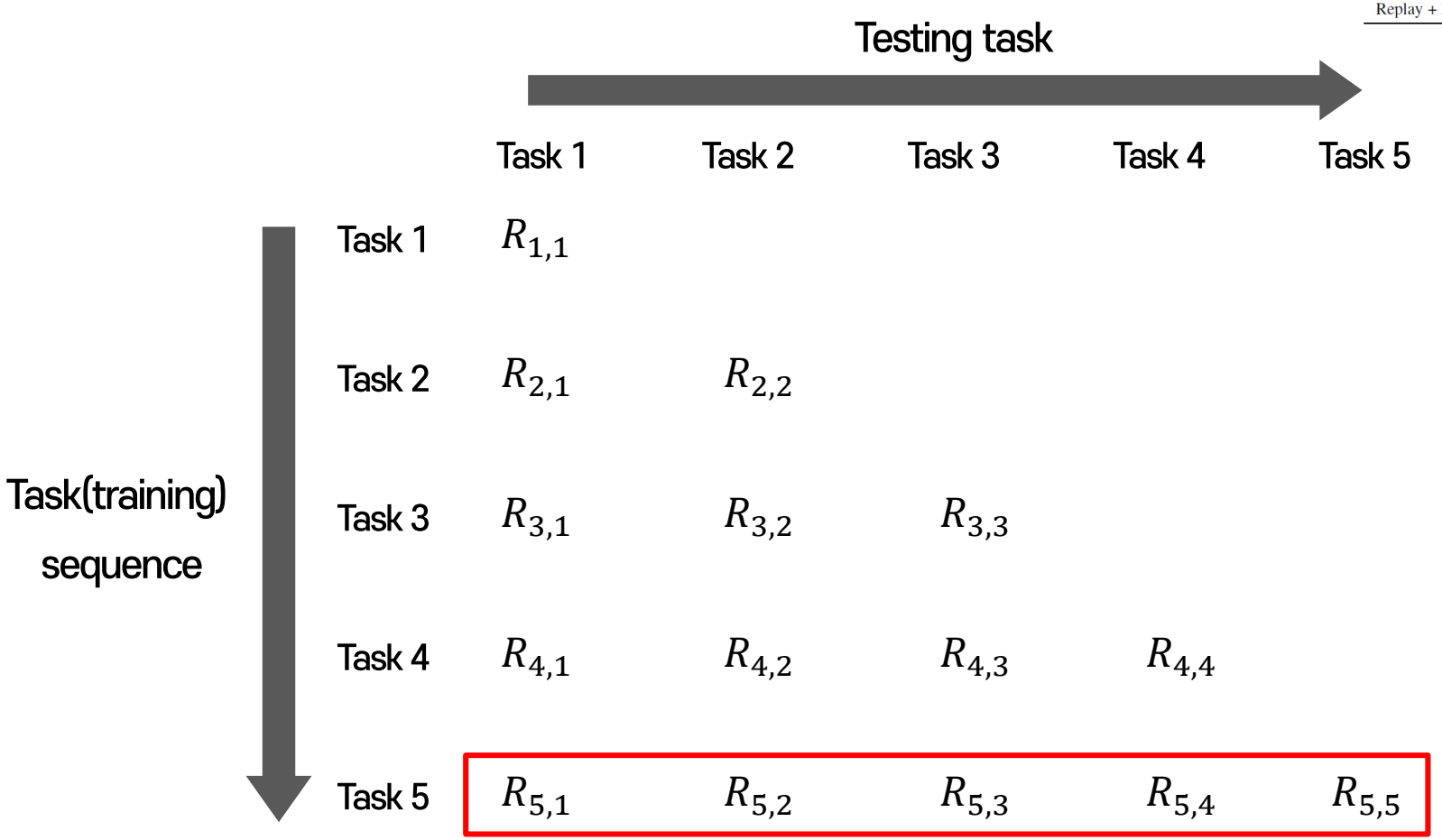
Approach	Method	Task-IL	Domain-IL	Class-IL
Baselines	*None – lower bound*	*87.19 (± 0.94)*	*59.21 (± 2.04)*	*19.90 (± 0.02)*
	Offline – upper bound	*99.66 (± 0.02)*	*98.42 (± 0.06)*	*97.94 (± 0.03)*
Task-specific	XdG	99.10 (± 0.08)	-	-
Regularization	EWC	98.64 (± 0.22)	63.95 (± 1.90)	20.01 (± 0.06)
	Online EWC	99.12 (± 0.11)	64.32 (± 1.90)	19.96 (± 0.07)
	SI	99.09 (± 0.15)	65.36 (± 1.57)	19.99 (± 0.06)
Replay	LwF	99.57 (± 0.02)	71.50 (± 1.63)	23.85 (± 0.44)
	DGR	99.50 (± 0.03)	95.72 (± 0.25)	90.79 (± 0.41)
	DGR+distill	99.61 (± 0.02)	96.83 (± 0.20)	91.79 (± 0.32)
Replay + Exemplars	iCaRL (budget = 2000)	-	-	94.57 (± 0.11)

Van de Ven, G. M., & Tolias, A. S. "Three scenarios for continual learning." (2018). Neural Information Processing Systems Workshops, 2018.

Evaluation metrics

Approach	Method	Task-IL	Domain-IL	Class-IL
Baselines	*None – lower bound*	*87.19 (± 0.94)*	*59.21 (± 2.04)*	*19.90 (± 0.02)*
	Offline – upper bound	*99.66 (± 0.02)*	*98.42 (± 0.06)*	*97.94 (± 0.03)*
Task-specific	XdG	99.10 (± 0.08)	-	-
Regularization	EWC	98.64 (± 0.22)	63.95 (± 1.90)	20.01 (± 0.06)
	Online EWC	99.12 (± 0.11)	64.32 (± 1.90)	19.96 (± 0.07)
	SI	99.09 (± 0.15)	65.36 (± 1.57)	19.99 (± 0.06)
Replay	LwF	99.57 (± 0.02)	71.50 (± 1.63)	23.85 (± 0.44)
	DGR	99.50 (± 0.03)	95.72 (± 0.25)	90.79 (± 0.41)
	DGR+distill	99.61 (± 0.02)	96.83 (± 0.20)	91.79 (± 0.32)
Replay + Exemplars	iCaRL (budget = 2000)	-	-	94.57 (± 0.11)

- $R_{m,n}$: m번째 task까지 학습한 모델의 n번째 task의 성능



Task 5까지 학습한 모델 성능

= 각 testing task 성능의 평균

https://www.cs.uic.edu/~liub/Part-1-continual-learning-slides.pdf

Approach : Regularization-based

- Elastic Weight Consolidation (EWC) [2017 PNAS]

- Old tasks에서 학습된 weights가 new task의 학습 시에 변화하는 정도를 제약하여 **catastrophic forgetting을 방지**하고자 함
- Regularization을 통해 penalty를 주지 않으면, task B에 적합하게(low error) 대부분의 weights가 이동하여 부적절
- 이전 task weights에서 이번 task에 중요한 weights를 산출하여 regularization을 주는 형태의 알고리즘 제안

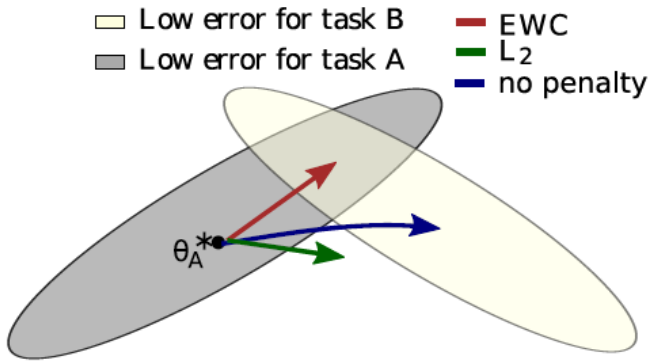


Figure 1: elastic weight consolidation (EWC) ensures task A is remembered whilst training on task B. Training trajectories are illustrated in a schematic parameter space, with parameter regions leading to good performance on task A (gray) and on task B (cream). After learning the first task, the parameters are at θ^*_A. If we take gradient steps according to task B alone (blue arrow), we will minimize the loss of task B but destroy what we have learnt for task A. On the other hand, if we constrain each weight with the same coefficient (green arrow) the restriction imposed is too severe and we can only remember task A at the expense of not learning task B. EWC, conversely, finds a solution for task B without incurring a significant loss on task A (red arrow) by explicitly computing how important weights are for task A.

Kirkpatrick, James, et al. "Overcoming catastrophic forgetting in neural networks." Proceedings of the national academy of sciences 114.13 (2017): 3521-3526.

Approach : Regularization-based

- Elastic Weight Consolidation (EWC) [2017 PNAS]

- Bayes' rule을 활용하여 우선 data(D)와 weights(θ)로 표현할 수 있음

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

- 첫번째 data(task) A에 대해 학습한 후의 posterior는 다음 data(task) B에 대한 prior으로 볼 수 있음

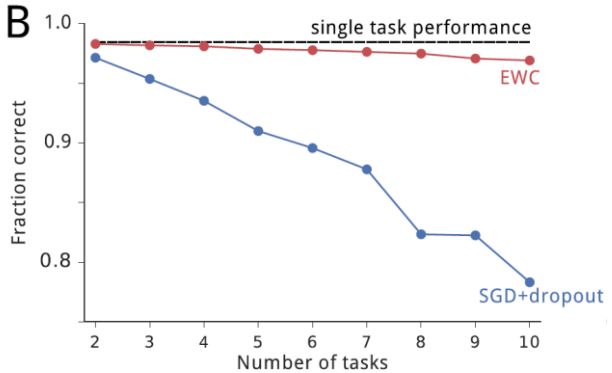
$$\log p(\theta|\mathcal{D}) = \log p(\mathcal{D}_B|\theta) + \log p(\theta|\mathcal{D}_A) - \log p(\mathcal{D}_B)$$

- Laplace approximation(가정 : all weights are Gaussian distribution)을 통해, 최종 loss function 도출

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

→ 이전 데이터의 중요도 설정

→ Fisher information matrix (이전 데이터와의 관계)



Kirkpatrick, James, et al. "Overcoming catastrophic forgetting in neural networks." Proceedings of the national academy of sciences 114.13 (2017): 3521-3526.
https://www.reddit.com/r/2D3DAI/comments/js69za/references_from_lecture_introduction_to_continual/

Approach : Replay-based

- Deep Generative Replay (DGR) [2017 NIPS]

- Replay 기반의 continual learning 방식은 task의 수가 많아질수록 보관해야 하는 train set의 크기가 커지므로 memory 이슈가 있음
- 한정된 자원을 활용하는 인간의 뇌 학습 방법을 살펴보면, 해마(hippocampus)는 빠른 속도로 단기 기억을 담당하고 여러 번 replay 된 후 신피질(neocortex)에 장기 기억으로 저장되는 방식
- 설명한 아이디어를 neural network에 **generative model**을 활용하여 **memory**를 **대체**하는 형식의 알고리즘

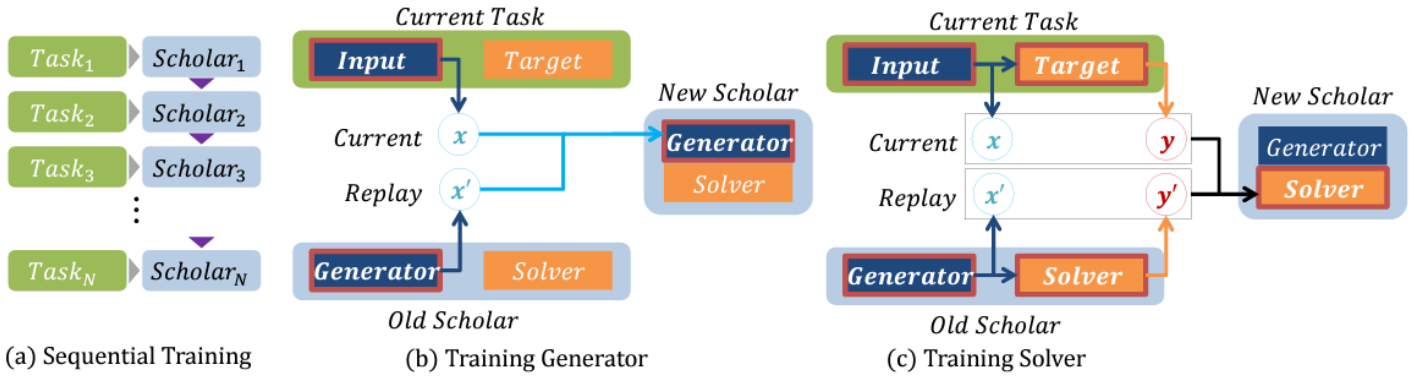


Figure 1: Sequential training of scholar models. (a) Training a sequence of scholar models is equivalent to continuous training of a single scholar while referring to its most recent copy. (b) A new generator is trained to mimic a mixed data distribution of real samples $\boldsymbol{x}$ and replayed inputs $\boldsymbol{x}'$ from previous generator. (c) A new solver learns from real input-target pairs $(\boldsymbol{x}, \boldsymbol{y})$ and replayed input-target pairs $(\boldsymbol{x}', \boldsymbol{y}')$, where replayed response $\boldsymbol{y}'$ is obtained by feeding generated inputs into previous solver.

Shin, Hanul, et al. "Continual learning with deep generative replay." Advances in neural information processing systems 30 (2017).

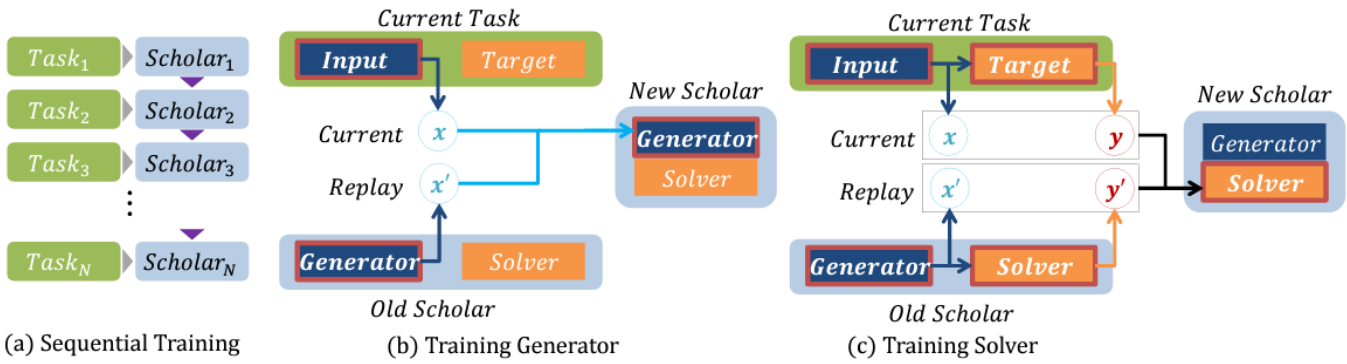
Approach : Replay-based

- Deep Generative Replay (DGR) [2017 NIPS]

- Task 마다 scholar 라는 것이 학습이 되고, 이 안에는 generator와 solver가 있음
- Generator는 input data와 유사한 data를 만드는 역할이고, 현재 task의 input(x)과 이전 tasks의 replay된 input(x')이 섞여서 **모든 task**의 input data를 반영하는 generator로 학습이 진행됨
- Solver는 각 task에서 주어진 문제를 풀어내는 역할이고, 현재 task의 (x, y)와 generator로부터 나온 (x', y')을 전부 사용하여 **이전 데이터**까지 반영하는 solver를 학습시킴

$$L_{train}(\theta_i) = r\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim D_i}[L(S(\boldsymbol{x};\theta_i),\boldsymbol{y})] + (1-r)\,\mathbb{E}_{\boldsymbol{x}'\sim G_{i-1}}[L(S(\boldsymbol{x}';\theta_i),S(\boldsymbol{x}';\theta_{i-1}))]$$

현재 task의 중요도



Shin, Hanul, et al. "Continual learning with deep generative replay." Advances in neural information processing systems 30 (2017).

Approach : Replay-based

- Deep Generative Replay (DGR) [2017 NIPS]

- Task 마다 scholar 라는 것이 학습이 되고, 이 안에는 generator와 solver가 있음
- Generator는 input data와 유사한 data를 만드는 역할이고, 현재 task의 input(x)과 이전 tasks의 replay된 input(x')이 섞여서 **모든 task**의 input data를 반영하는 generator로 학습이 진행됨
- Solver는 각 task에서 주어진 문제를 풀어내는 역할이고, 현재 task의 (x, y)와 generator로부터 나온 (x', y')을 전부 사용하여 **이전 데이터**까지 반영하는 solver를 학습시킴

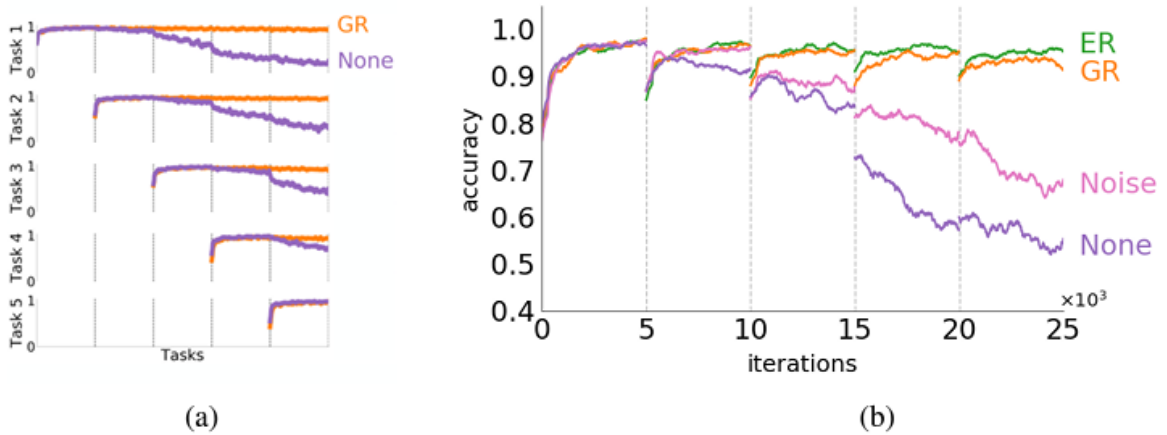


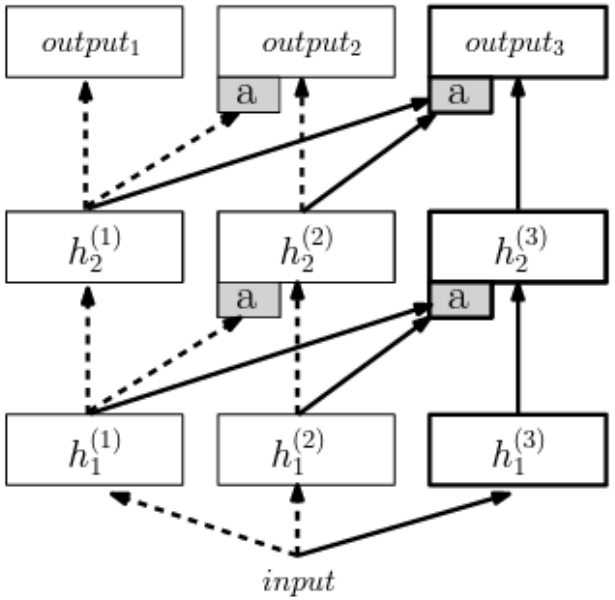
Figure 2: Results on MNIST pixel permutation tasks. (a) Test performances on each task during sequential training. Performances for previous tasks dropped without replaying real or meaningful fake data. (b) Average test accuracy on learnt tasks. Higher accuracy is achieved when the replayed inputs better resembled real data.

Shin, Hanul, et al. "Continual learning with deep generative replay." Advances in neural information processing systems 30 (2017).

Approach : Architecture-based

- Progressive Neural Networks [2016 Arxiv]

- Catastrophic forgetting 문제를 해결하기 위해, **model 구조 관점**(lateral connection)에서 knowledge transfer을 가능하게 하는 방법을 제안
- Task 마다 1개 column 형태의 neural network(fully connected layer, number of layer : L)로 구성됨
- Task가 K이고 U가 old tasks의 학습된 weight matrix 일 때, h는 다음과 같음



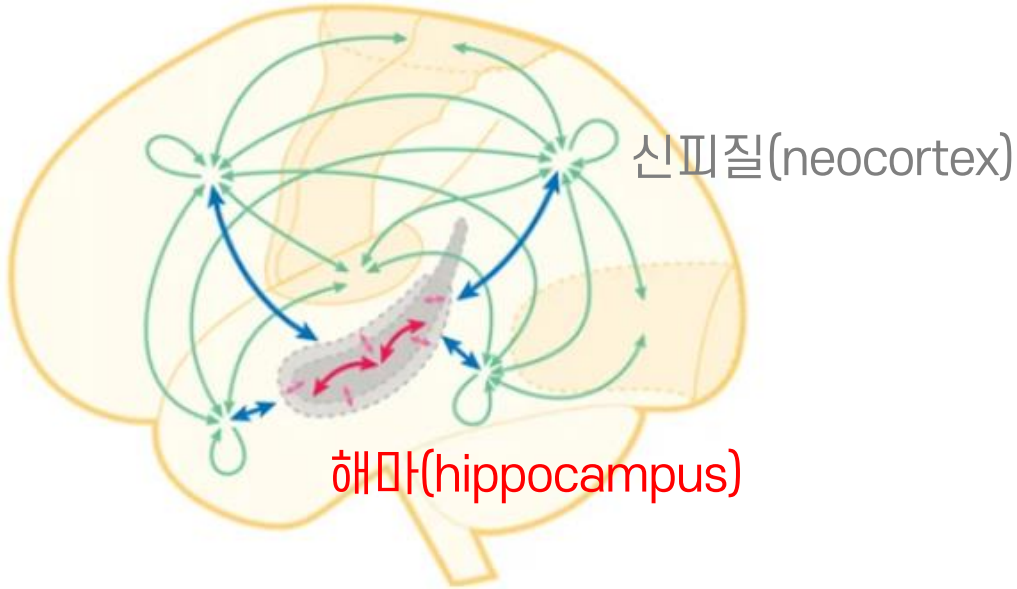
$$h_i^{(k)} = f\left(W_i^{(k)} h_{i-1}^{(k)} + \sum_{j<k} U_i^{(k:j)} h_{i-1}^{(j)}\right)$$

Rusu, Andrei A., et al. "Progressive neural networks." arXiv preprint arXiv:1606.04671 (2016).

Recent approach

- DualNet [2021 NeurIPS]

- DGR (2017 NIPS)에도 소개된 개념인 장/단기 기억에 관한 인간(뇌)의 학습 방식(complementary learning systems theory)을 소개
- 해마(hippocampus)는 빠르고 개별적인 단기 기억을 담당하고, 신피질(neocortex)은 상대적으로 느리며 장기 기억(knowledge structure)을 담당함
- 두 개의 부분이 **상호 작용**하면서 knowledge acquisition과 consolidation을 확보



Pham, Quang, Chenghao Liu, and Steven Hoi. "Dualnet: Continual learning, fast and slow." Advances in Neural Information Processing Systems 34 (2021): 16131-16144.
https://slideslive.com/38968809

Recent approach

- DualNet [2021 NeurIPS]

- 해마와 신피질의 역할은 각각 fast net과 slow net의 neural network 형태로 나타내며, backbone은 ResNet을 사용
- Slow net은 **self-supervised learning**(Barlow Twins, 2021 PMLR) 방식을 통해 데이터의 일반적인 feature를 추출하여, fast net에게 feature adaptation 방식으로 전달
- Fast net은 전달받은 feature와 현재 task의 input data 및 label을 활용하여 **supervised learning**을 통해 학습

→ 이전 tasks의 episodic memory를 활용하여 replay 방식으로 성능 개선

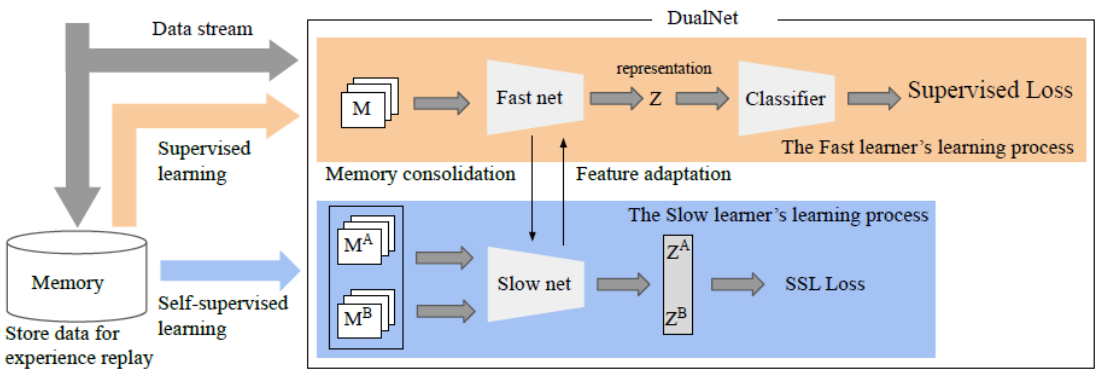


Figure 1: Overview of the DualNet' architecture. DualNet consists of (i) a slow learner (in blue) that learns representation by optimizing an SSL loss using samples from the memory, and (ii) a fast learner (in orange) that adapts the slow net's representation for quick knowledge acquisition of labeled data. Both learners can be trained synchronously.

Pham, Quang, Chenghao Liu, and Steven Hoi. "Dualnet: Continual learning, fast and slow." Advances in Neural Information Processing Systems 34 (2021): 16131-16144.

Recent approach

- DualNet [2021 NeurIPS]

- Split miniImageNet-TA에서 TA는 task-aware로 기존의 Task-IL 방식을 의미
- Split miniImageNet-TF에서 TF는 task-free로 기존의 Class-IL 방식을 의미

Forgetting measure (FM) :

$$f_j^k = \max_{l \in \{1,\cdots,k-1\}} a_{l,j} - a_{k,j} \ , \quad \forall j < k \ .$$

Table 1: Evaluation metrics on the Split miniImageNet and CORE50 benchmarks. All methods use an episodic memory of 50 samples per task in the TA setting, and 100 samples per class in the TF setting. The "Aug" suffix denotes using data augmentation during training

Method	Split miniImageNet-TA			Split miniImageNet-TF		
	ACC(↑)	FM(↓)	LA(↑)	ACC(↑)	FM(↓)	LA(↑)
ER	58.24±0.78	9.22±0.78	65.36±0.71	25.12±0.99	28.56±1.10	49.04±1.56
ER-Aug	59.80±1.51	4.68±1.21	58.94±0.69	27.94±2.44	29.36±3.23	54.02±1.02
DER++	62.32±0.78	7.00±0.81	67.30±0.57	27.16±1.99	34.56±2.48	59.54±1.53
DER++-Aug	63.48±0.98	4.01±1.21	62.17±0.52	28.26±1.81	36.70±1.85	62.70±0.41
CTN	65.82±0.59	**3.02±1.13**	67.43±1.37	N/A	N/A	N/A
CTN-Aug	68.04±1.23	3.94±0.98	69.84±0.78	N/A	N/A	N/A
DualNet	**73.20±0.68**	**3.86±1.01**	**74.12±0.12**	**36.86±1.36**	**28.63±2.26**	**63.46±1.97**
Method	CORE50-TA			CORE50-TF		
	ACC(↑)	FM(↓)	LA(↑)	ACC(↑)	FM(↓)	LA(↑)
ER	41.72±1.30	9.10±0.80	48.18±0.81	21.80±0.70	14.42±1.10	33.94±1.49
ER-Aug	44.16±2.05	5.72±0.02	47.83±1.61	25.34±0.74	15.28±0.63	37.94±0.91
DER	46.62±0.46	4.66±0.46	48.32±0.69	22.84±0.84	13.10±0.40	34.50±0.81
DER++-Aug	45.12±0.68	5.02±0.98	47.67±0.08	28.10±0.80	10.43±2.10	36.16±0.19
CTN	54.17±0.85	5.50±1.10	55.32±0.34	N/A	N/A	N/A
CTN-Aug	53.40±1.37	6.18±1.61	55.40±1.47	N/A	N/A	N/A
DualNet	**57.64±1.36**	**4.43±0.82**	**58.86±0.66**	**38.76±1.52**	**8.06±0.43**	**40.00±1.67**

Table 2: DualNet's performance under different slow learner objective and optimizers on the Split miniImageNet-TA benchmark

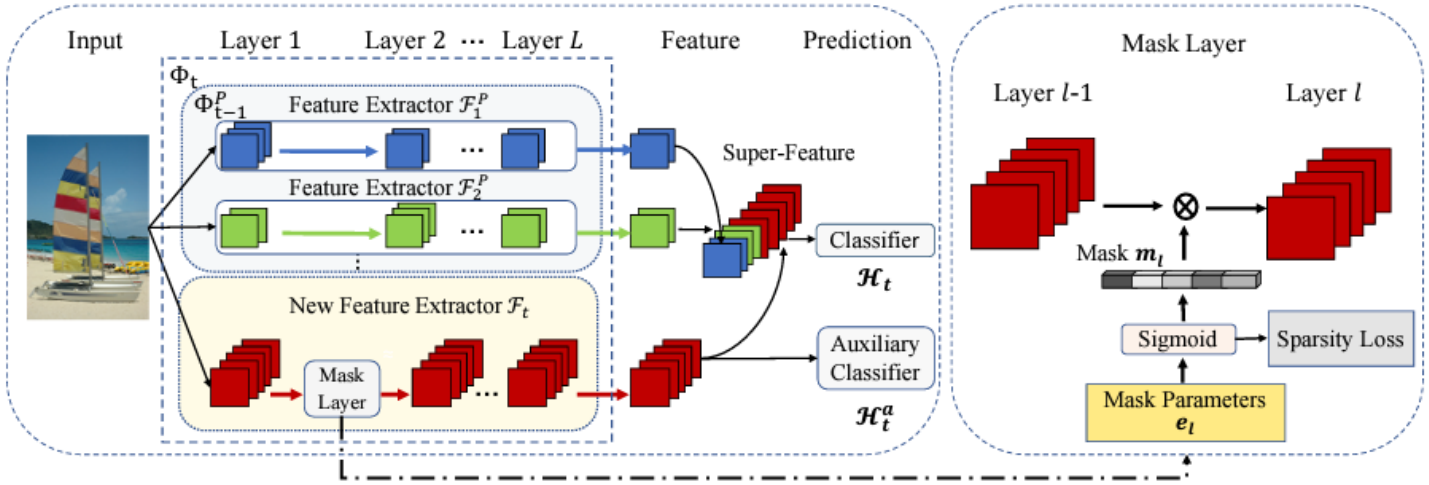
Objective	SGD			Look-ahead		
	ACC(↑)	FM(↓)	LA(↑)	ACC(↑)	FM(↓)	LA(↑)
Barlow Twins	64.20±2.37	4.79±1.19	64.83±1.67	**73.20±0.68**	**3.86±1.01**	**74.12±0.12**
SimCLR	**71.49±1.01**	**4.23±0.46**	**72.64±1.20**	72.13±0.44	4.13±0.52	73.09±0.16
SimSiam	70.55±0.98	4.93±1.31	71.90±0.65	71.94±0.64	4.21±0.28	72.93±0.38
BYOL	69.76±2.12	**4.23±1.41**	70.33±0.87	71.73±0.47	**3.96±0.62**	72.06±0.28
Classification	68.50±1.67	5.53±1.67	**72.93±1.10**	70.96±1.08	6.33±0.28	73.92±1.14

Pham, Quang, Chenghao Liu, and Steven Hoi. "Dualnet: Continual learning, fast and slow." Advances in Neural Information Processing Systems 34 (2021): 16131-16144.

Recent approach

- Dynamic Expandable Representation (DER) [2021 CVPR]

- T시점의 task를 학습할 때, input data(x)에 대해 F(t)가 학습되고 t-1 까지의 feature extractor는 weights 고정
- Input data(x)를 각 extractor에 통과시킨 후 모든 t에 대해 feature를 통합한 것을 u라고 함 $\boldsymbol{u} = \Phi_t(\boldsymbol{x}) = [\Phi_{t-1}(\boldsymbol{x}), \mathcal{F}_t(\boldsymbol{x})]$
- 시점 T에서 class를 분류하는 classifier H를 u를 활용하여 학습함 $p_{\mathcal{H}_t}(\boldsymbol{y}|\boldsymbol{x}) = \mathrm{Softmax}(\mathcal{H}_t(\boldsymbol{u}))$
- Train loss는 **class 분류 loss**와 old/new task를 분류하는 auxiliary loss를 더하여 계산 $\mathcal{L}_{\mathrm{ER}} = \mathcal{L}_{\mathcal{H}_t} + \lambda_a \mathcal{L}_{\mathcal{H}_t^a}$
- 효율적인 모델을 위해 feature의 channel 수를 줄여주는 sparsity loss 추가 $\mathcal{L}_{\mathrm{DER}} = \mathcal{L}_{\mathcal{H}_t} + \lambda_a \mathcal{L}_{\mathcal{H}_t^a} + \lambda_s \mathcal{L}_S$



Yan, Shipeng, Jiangwei Xie, and Xuming He. "DER: Dynamically Expandable Representation for Class Incremental Learning." 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society, 2021.

Conclusion

1) 본 세미나에서는 현실 문제 상황에서 continual learning에 대한 **필요성** 및 주요 용어를 정리
2) 특히, neural network 기반의 모델에서 새로운 data(task)로 network를 학습시키면 **catastrophic forgetting** 문제가 발생하기 때문에, 이를 해결하고자 많은 연구들이 제시되고 있음
3) 연구 초기에는 regularization-based 방법론이 연구되었고 최근에는 다양한 방법을 혼합하는 방식이 주를 이루고 있고 (e.g., replay + knowledge distillation), 뇌 과학 개념을 도입하기도 함 (CLS theory)
4) 다만, 현실의 continual learning 상황에 비해 논문에서 제시하고 있는 문제 상황은 비교적 단순. **다양한 scenario**를 **작성**하고 이에 대해 검증하는 식으로 발전 기대

Thank you

Seokho Moon

danny232@korea.ac.kr

School of Industrial and Management Engineering, Korea University