

# Generative Replay for Continual Learning

안시후

2024.09.06

# 발표자 소개



## ❖ 안시후 (Sihu Ahn)

- Data Mining & Quality Analytics Lab (김성범 교수님)
- 석박사통합과정 (2021.3 ~)

## ❖ 관심 연구 분야

- Time Series Data Analysis
- Diffusion-based Generative Replay for Continual Learning
- Computer Vision (Action Recognition)

## ❖ E-mail

- [sihuahn@korea.ac.kr](mailto:sihuahn@korea.ac.kr)

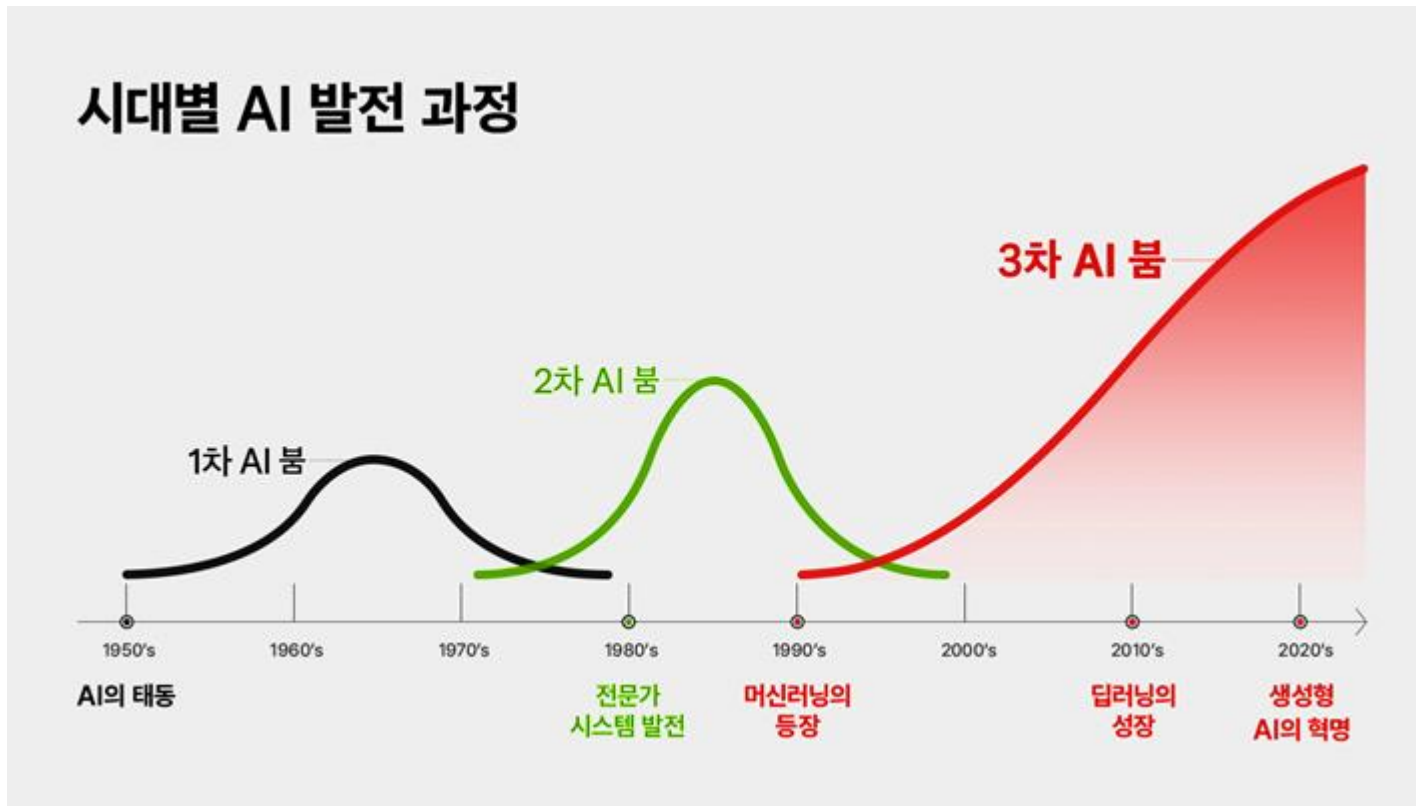
# Seminar Outline

- ❖ **Introduction to Continual Learning and Replay Methods**
- ❖ **Applications of Generative Replay in Image Data**
  - Continual learning with deep generative replay
  - Continual learning with deep diffusion-based generative replay
- ❖ **Summary**

# 1. Introduction to Continual Learning and Replay Methods

## ❖ AI의 발전

- 기술 발전에 따라 다양한 분야에서 우수한 모델 출현
- 실제 서비스로 출시되고 있는 상황



TESLA

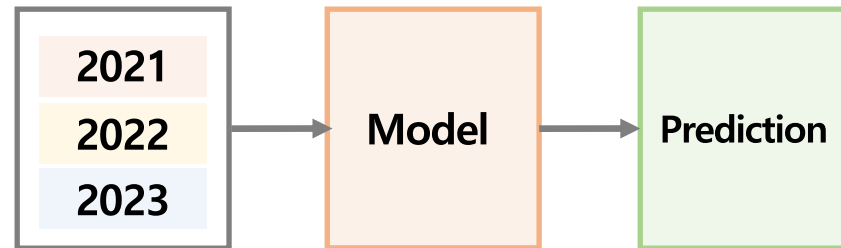


# 1. Introduction to Continual Learning and Replay Methods

---

## ❖ 실제 환경에 따른 한계점

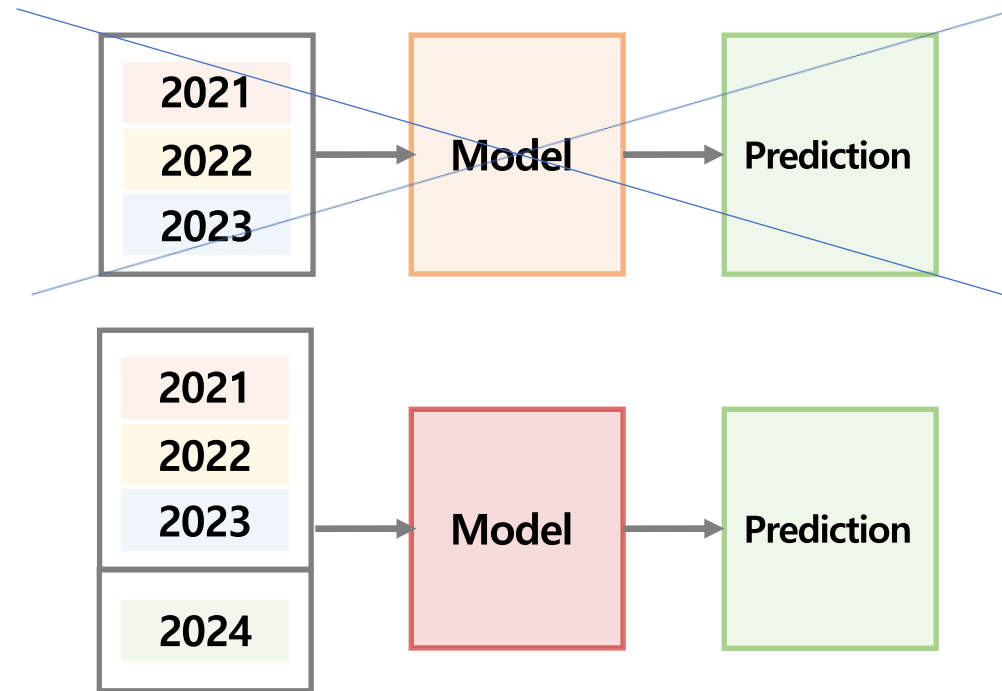
- 시간이 지남에 따라 새로운 데이터를 업데이트 필요
  - ✓ 2023년까지의 데이터만으로 학습된 모델에서 2024년 정보를 기대하기는 어려움



# 1. Introduction to Continual Learning and Replay Methods

## ❖ 실제 환경에 따른 한계점

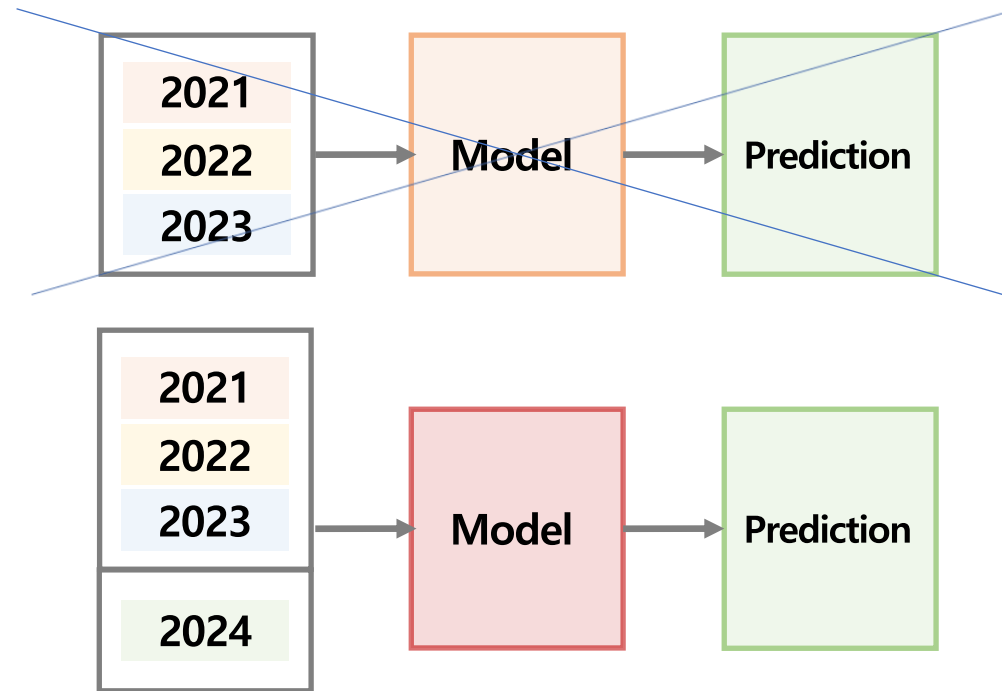
- 시간이 지남에 따라 새로운 데이터를 업데이트 필요
  - ✓ 2023년까지의 데이터만으로 학습된 모델에서 2024년 정보를 기대하기는 어려움
- 일반적으로 기존 데이터에 새로운 데이터를 추가하여 처음부터 모델 학습



# 1. Introduction to Continual Learning and Replay Methods

## ❖ 실제 환경에 따른 한계점

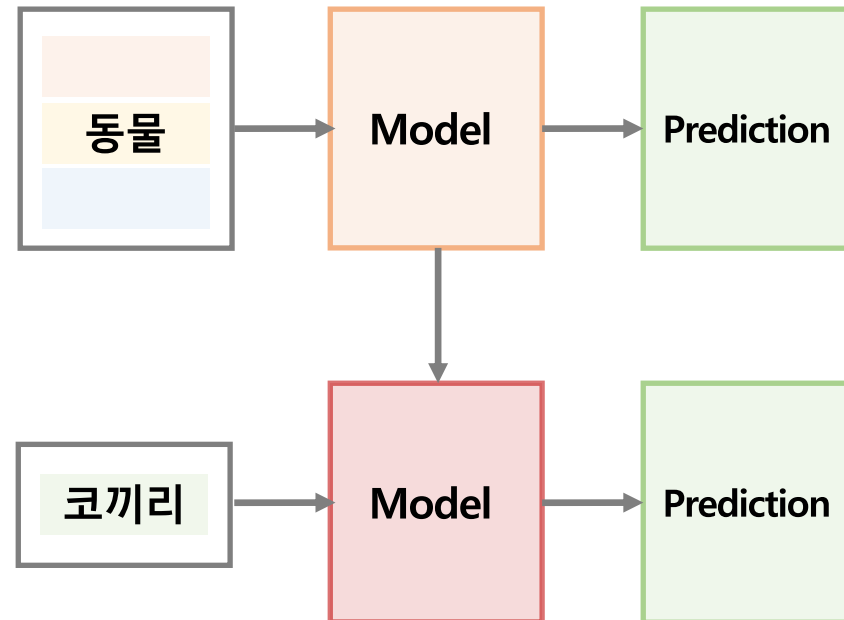
- 시간이 지남에 따라 새로운 데이터를 업데이트 필요
  - ✓ 2023년까지의 데이터만으로 학습된 모델에서 2024년 정보를 기대하기는 어려움
- 일반적으로 기존 데이터에 새로운 데이터를 추가하여 처음부터 모델 학습
  - ✓ 기존 모델을 버리고, 데이터를 저장해두어야 한다는 문제 발생



# 1. Introduction to Continual Learning and Replay Methods

## ❖ 기존 데이터 없이 잘 학습된 모델을 활용 할 수 있을까?

- 학습된 모델을 이용하는 일반적인 방법
  - ✓ 전이학습 (Transfer Learning): 일반적인 지식을 새로운 작업에 적용
    - 기존 학습 모델이 매우 큰 모델인 경우에 사용, 도메인 차이 문제 발생 및 모델 파라미터를 얼마나 고정 시킬지 고려 필요
  - ✓ 미세조정 (Fine-Tuning): 새로운 작업에 모델 성능 최적화
    - 기존 데이터의 정보를 잊는 Catastrophic Forgetting 발생





# 1. Introduction to Continual Learning and Replay Methods

## ❖ 기존 데이터 없이 잘 학습된 모델을 활용 할 수 있을까?

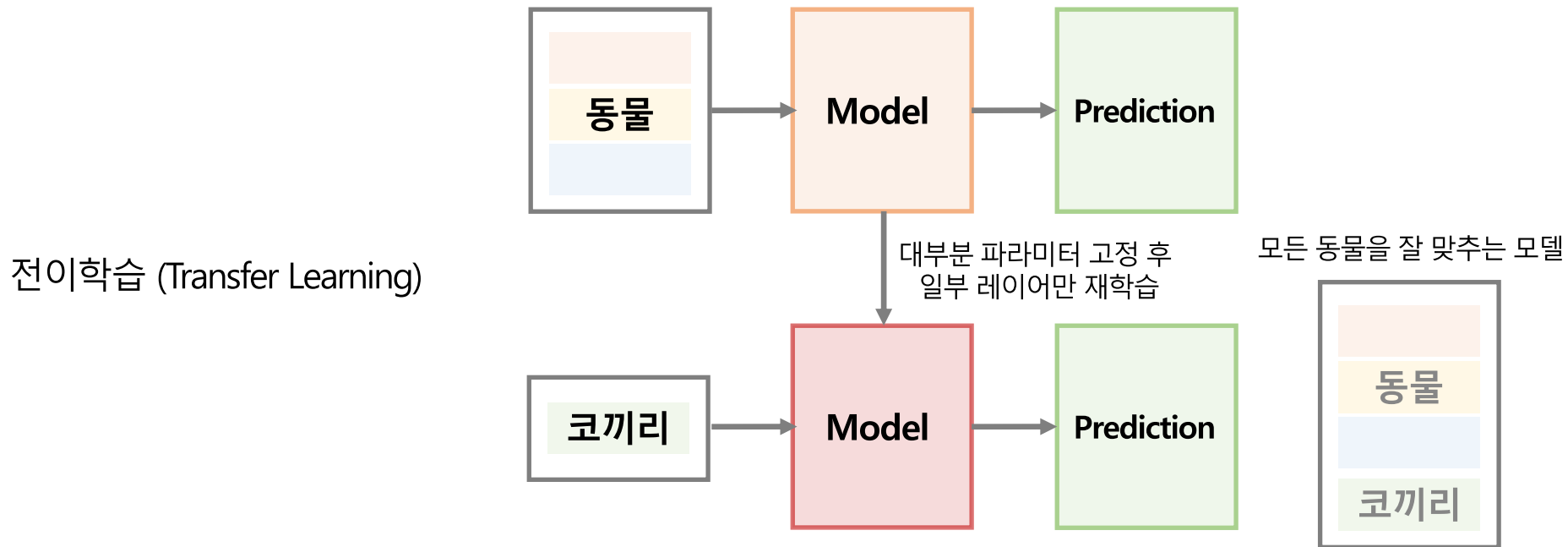
### • 학습된 모델을 이용하는 일반적인 방법

✓ 전이학습 (Transfer Learning): 일반적인 지식을 새로운 작업에 적용

→ 기존 학습 모델이 매우 큰 모델인 경우에 사용, 도메인 차이 문제 발생 및 모델 파라미터를 얼마나 고정 시킬지 고려 필요

✓ 미세조정 (Fine-Tuning): 새로운 작업에 모델 성능 최적화

→ 기존 데이터의 정보를 잊는 Catastrophic Forgetting 발생



# 1. Introduction to Continual Learning and Replay Methods

## ❖ 기존 데이터 없이 잘 학습된 모델을 활용 할 수 있을까?

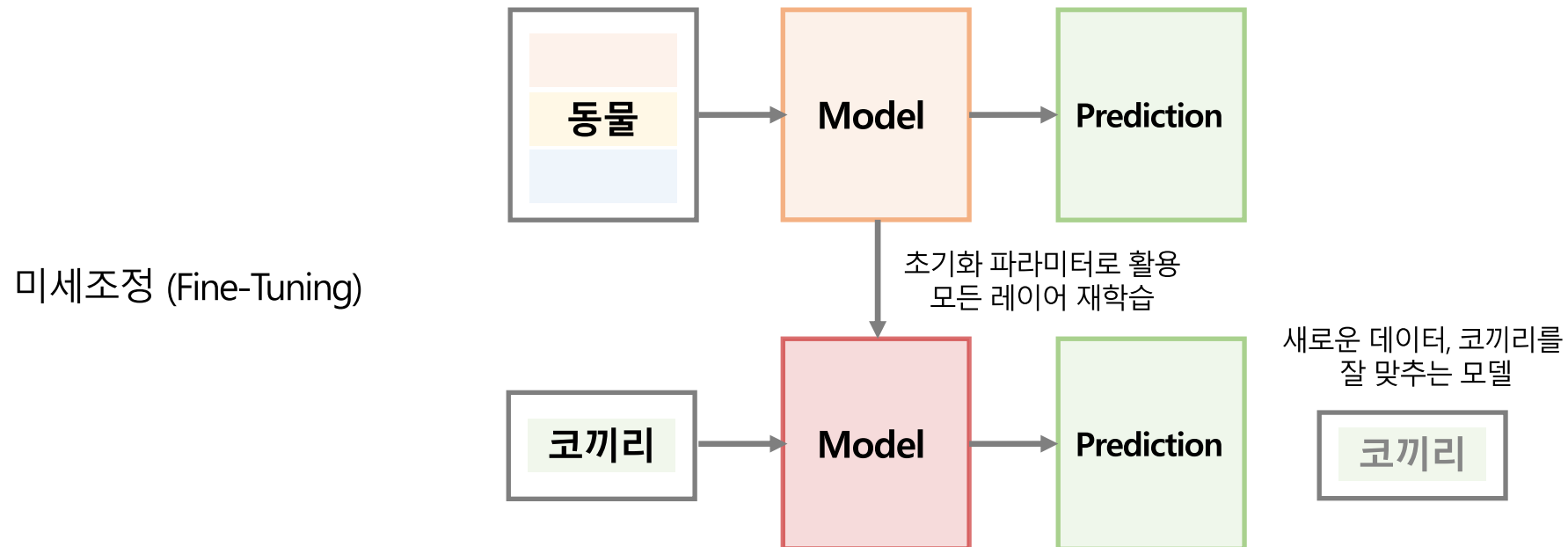
### • 학습된 모델을 이용하는 일반적인 방법

✓ 전이학습 (Transfer Learning): 일반적인 지식을 새로운 작업에 적용

→ 기존 학습 모델이 매우 큰 모델인 경우에 사용, 도메인 차이 문제 발생 및 모델 파라미터를 얼마나 고정 시킬지 고려 필요

✓ 미세조정 (Fine-Tuning): 새로운 작업에 모델 성능 최적화

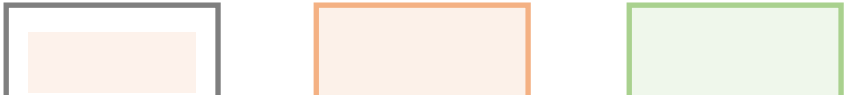
→ 기존 데이터의 정보를 잊는 Catastrophic Forgetting 발생



# 1. Introduction to Continual Learning and Replay Methods

## ❖ 기존 데이터 없이 잘 학습된 모델을 활용 할 수 있을까?

- 학습된 모델을 이용하는 일반적인 방법
  - ✓ 전이학습 (Transfer Learning): 일반적인 지식을 새로운 작업에 적용
    - 기존 학습 모델이 매우 큰 모델인 경우에 사용, 도메인 차이 문제 발생 및 모델 파라미터를 얼마나 고정 시킬지 고려 필요
  - ✓ 미세조정 (Fine-Tuning): 새로운 작업에 모델 성능 최적화
    - 기존 데이터의 정보를 잊는 Catastrophic Forgetting 발생



기존 데이터와 신규 데이터 모두 잘 맞춰야 하는 상황이라면?

기존 학습된 모델이 크지 않거나 도메인 차이가 있다면?

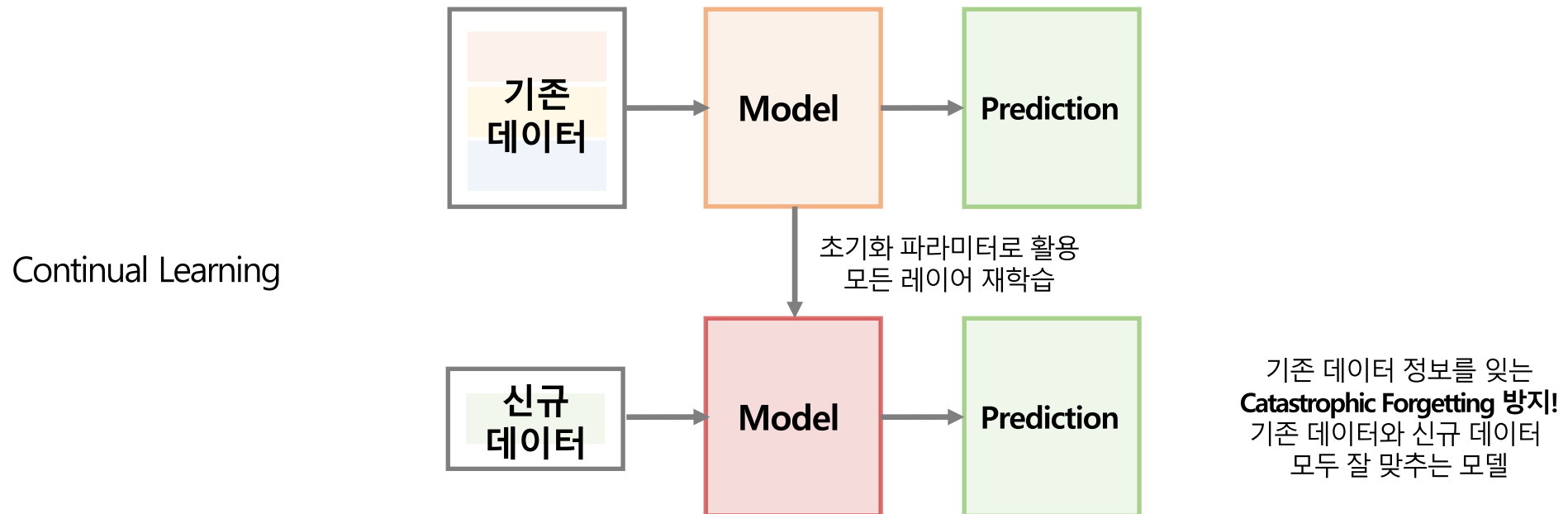


기존 데이터를 보관 할 수 없어 삭제된 경우라면?

# 1. Introduction to Continual Learning and Replay Methods

## ❖ Continual Learning의 정의와 중요성

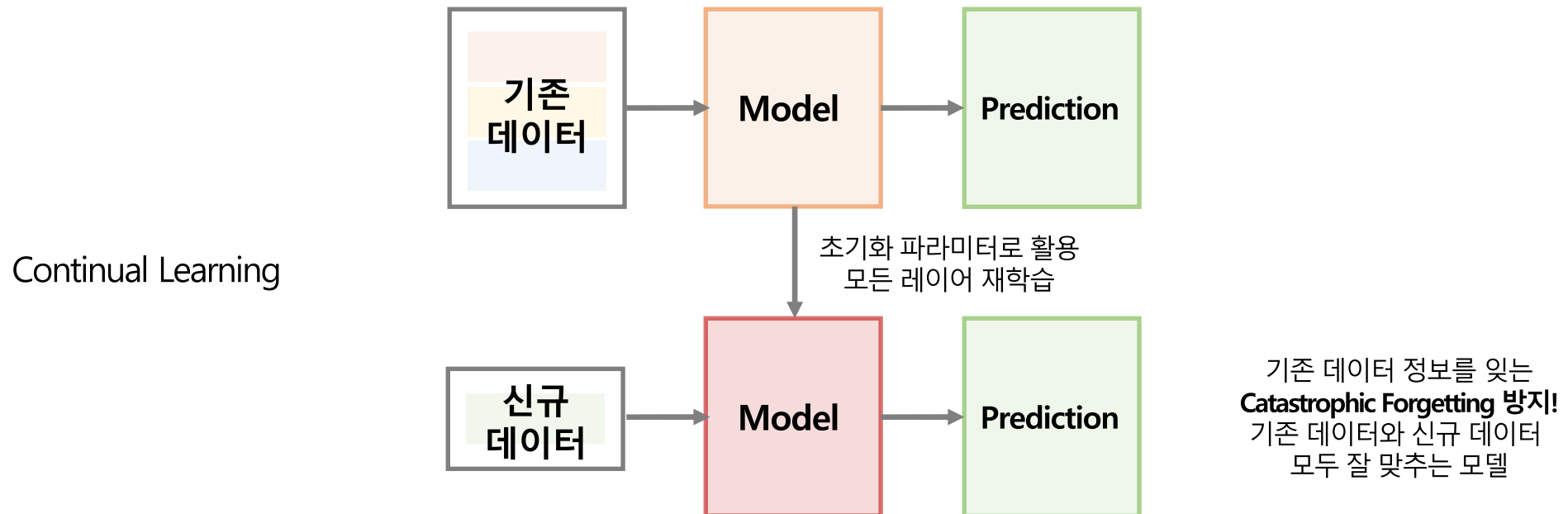
- 데이터를 지속적으로 학습하면서도 이전에 학습한 내용을 잃지 않고 유지하는 학습 방법
  - ✓ 모델이 변화하는 데이터 환경에서 지속적으로 업데이트하여 데이터가 점진적으로 추가 되거나 변경 될 때 유용 / 다른 Class, Domain 등
  - ✓ Fine-tuning을 기준으로 Catastrophic Forgetting을 해결하는 것이 목표 (Transfer learning은 유사한 데이터 특성 등 특정 조건 필요)
  - ✓ 기존 데이터를 보관 할 수 없는 경우를 생각하여 기존 데이터의 일부만 활용하거나 활용하지 않음



# 1. Introduction to Continual Learning and Replay Methods

## ❖ Continual Learning의 정의와 중요성

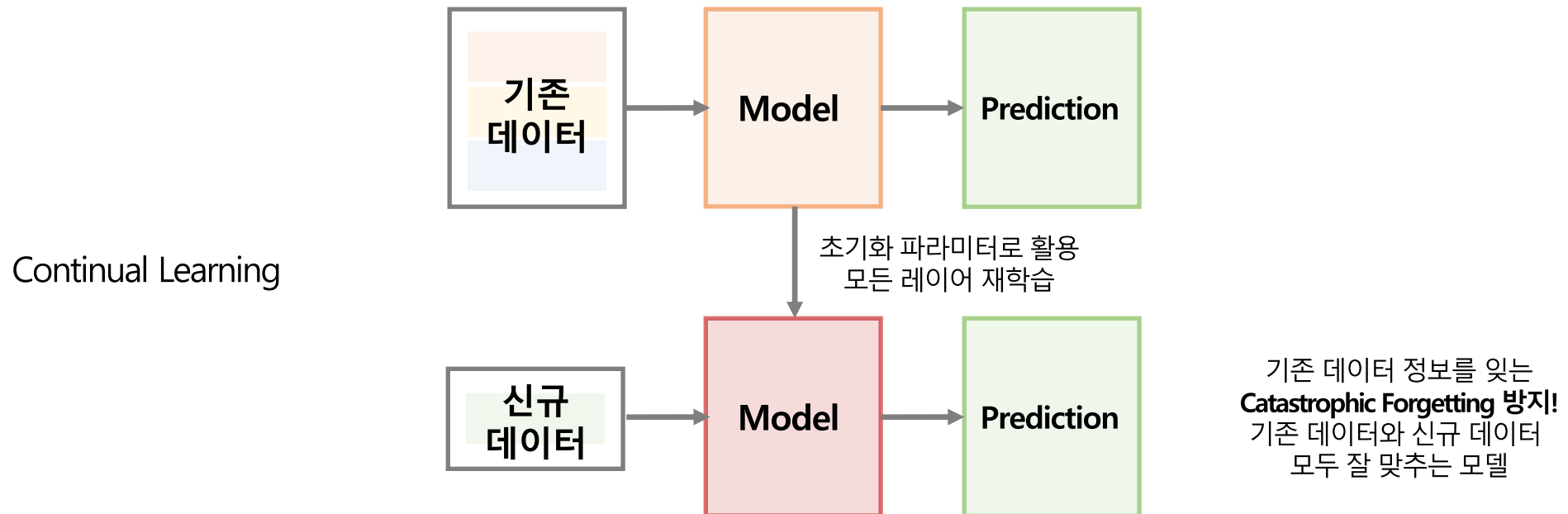
- 데이터를 지속적으로 학습하면서도 이전에 학습한 내용을 잃지 않고 유지하는 학습 방법
  - ✓ 모델이 변화하는 데이터 환경에서 지속적으로 업데이트하여 데이터가 점진적으로 추가 되거나 변경 될 때 유용 / 다른 Class, Domain 등
  - ✓ Fine-tuning을 기준으로 **Catastrophic Forgetting**을 해결하는 것이 목표 (Transfer learning은 유사한 데이터 특성 등 특정 조건 필요)
  - ✓ 기존 데이터를 보관 할 수 없는 경우를 생각하여 기존 데이터의 일부만 활용하거나 활용하지 않음



# 1. Introduction to Continual Learning and Replay Methods

## ❖ Catastrophic Forgetting을 해결하기 위한 두가지 관점

- 모델 관점
  - ✓ 가중치 정규화를 통해 기존 작업에서 중요했던 파라미터를 크게 변하지 않도록 제약을 주는 방식
- 데이터 관점
  - ✓ Memory Buffer에 데이터 일부를 재사용하거나 데이터를 생성하는 방식



# 1. Introduction to Continual Learning and Replay Methods

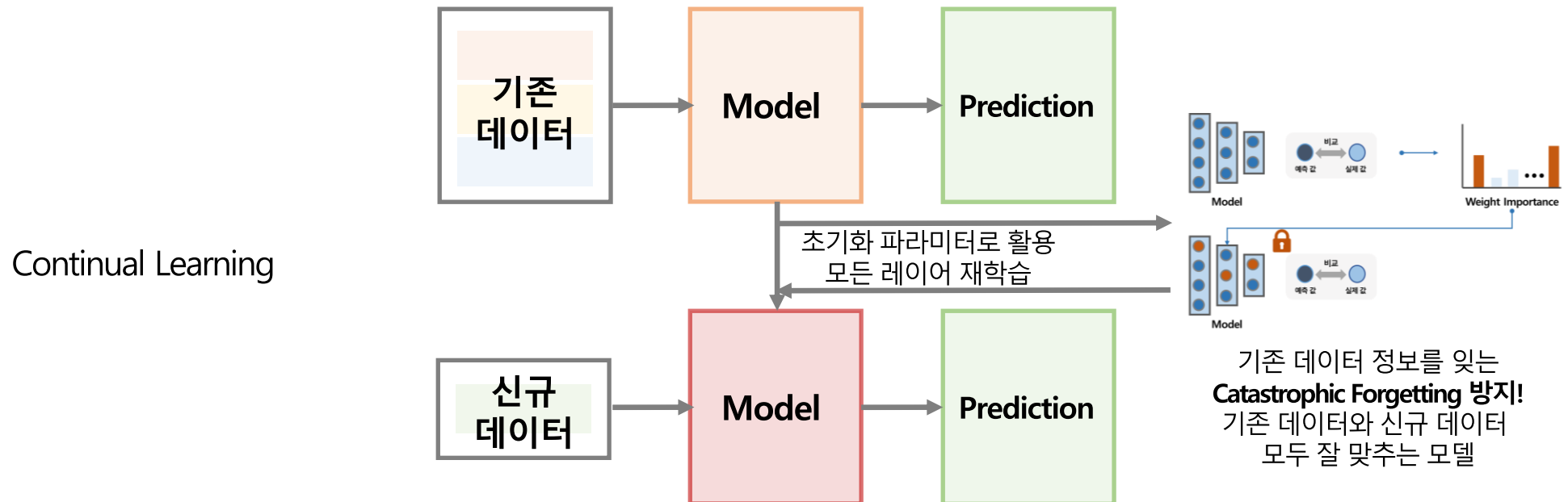
## ❖ Catastrophic Forgetting을 해결하기 위한 두가지 관점

### • 모델 관점

- ✓ 가중치 정규화를 통해 기존 작업에서 중요했던 파라미터를 크게 변하지 않도록 제약을 주는 방식

### • 데이터 관점

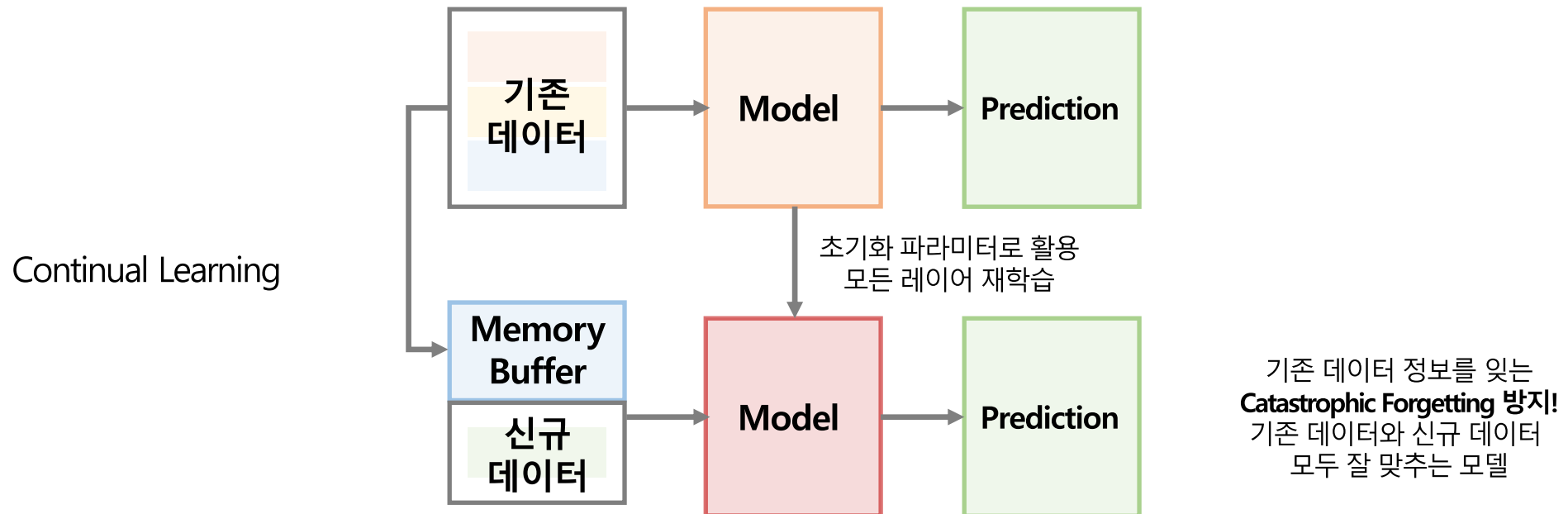
- ✓ Memory Buffer에 데이터 일부를 재사용하거나 데이터를 생성하는 방식



# 1. Introduction to Continual Learning and Replay Methods

## ❖ Catastrophic Forgetting을 해결하기 위한 두가지 관점

- 모델 관점
  - ✓ 가중치 정규화를 통해 기존 작업에서 중요했던 파라미터를 크게 변하지 않도록 제약을 주는 방식
- 데이터 관점
  - ✓ Memory Buffer에 데이터 일부를 재사용하거나 데이터를 생성하는 방식

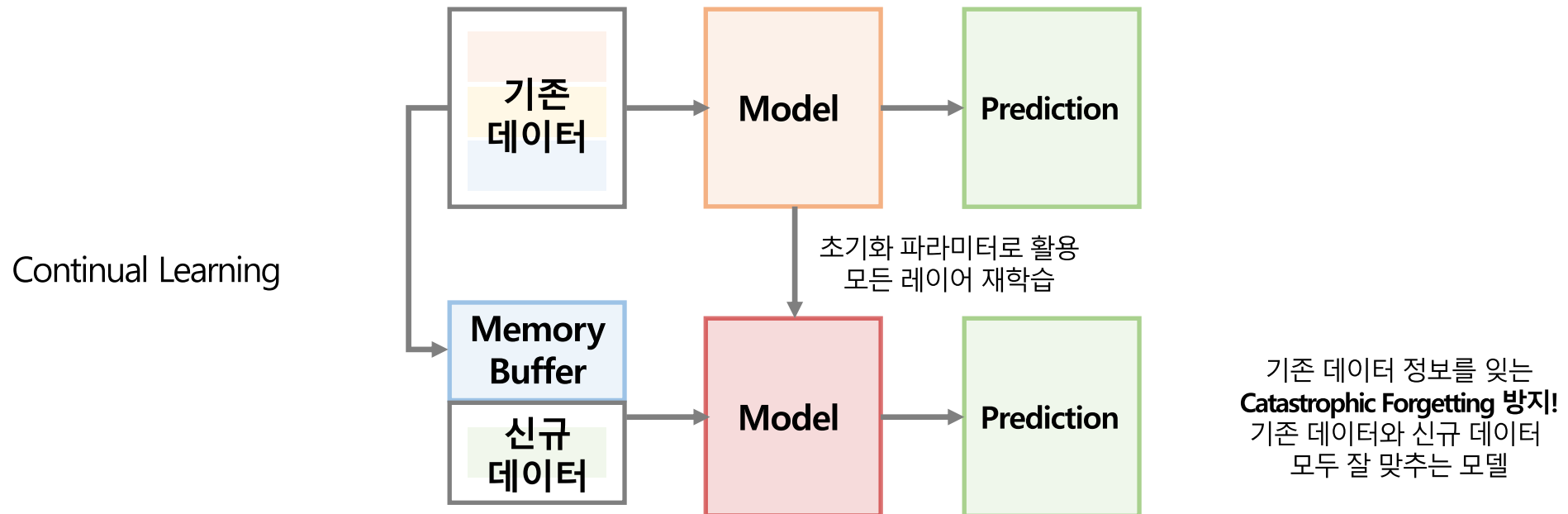




# 1. Introduction to Continual Learning and Replay Methods

## ❖ 데이터 관점에서의 Continual Learning

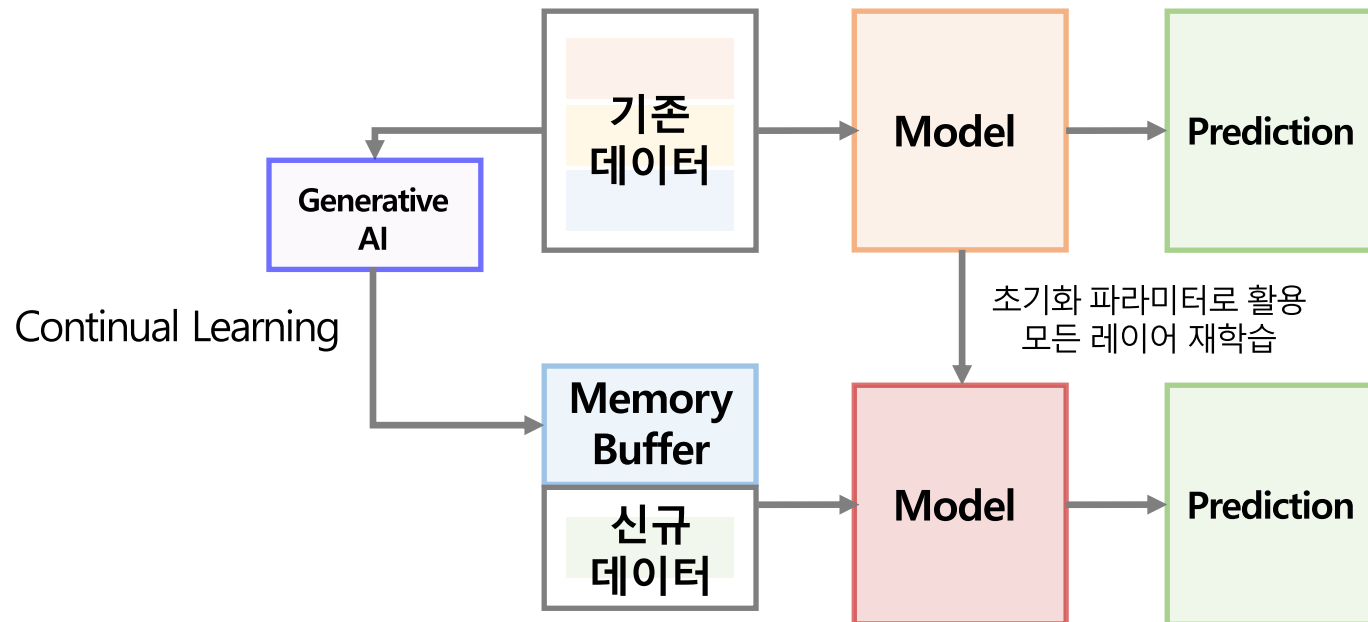
- 데이터 재사용
  - ✓ 간단하게 일부 기존 데이터를 사용하는 방식 → 현실적인 상황을 고려하여 일부 데이터만을 이용
- 데이터 생성
  - ✓ 기존 데이터로부터 학습된 생성형 모델을 이용하여 추가 학습시 사용될 데이터를 생성 → 데이터 일반화, 보안, 메모리 측면에서 장점 존재



# 1. Introduction to Continual Learning and Replay Methods

## ❖ 데이터 관점에서의 Continual Learning

- 데이터 재사용
  - ✓ 간단하게 일부 기존 데이터를 사용하는 방식 → 현실적인 상황을 고려하여 일부 데이터만을 이용
- 데이터 생성
  - ✓ 기존 데이터로부터 학습된 생성형 모델을 이용하여 추가 학습시 사용될 데이터를 생성 → 데이터 일반화, 보안, 메모리 측면에서 장점 존재

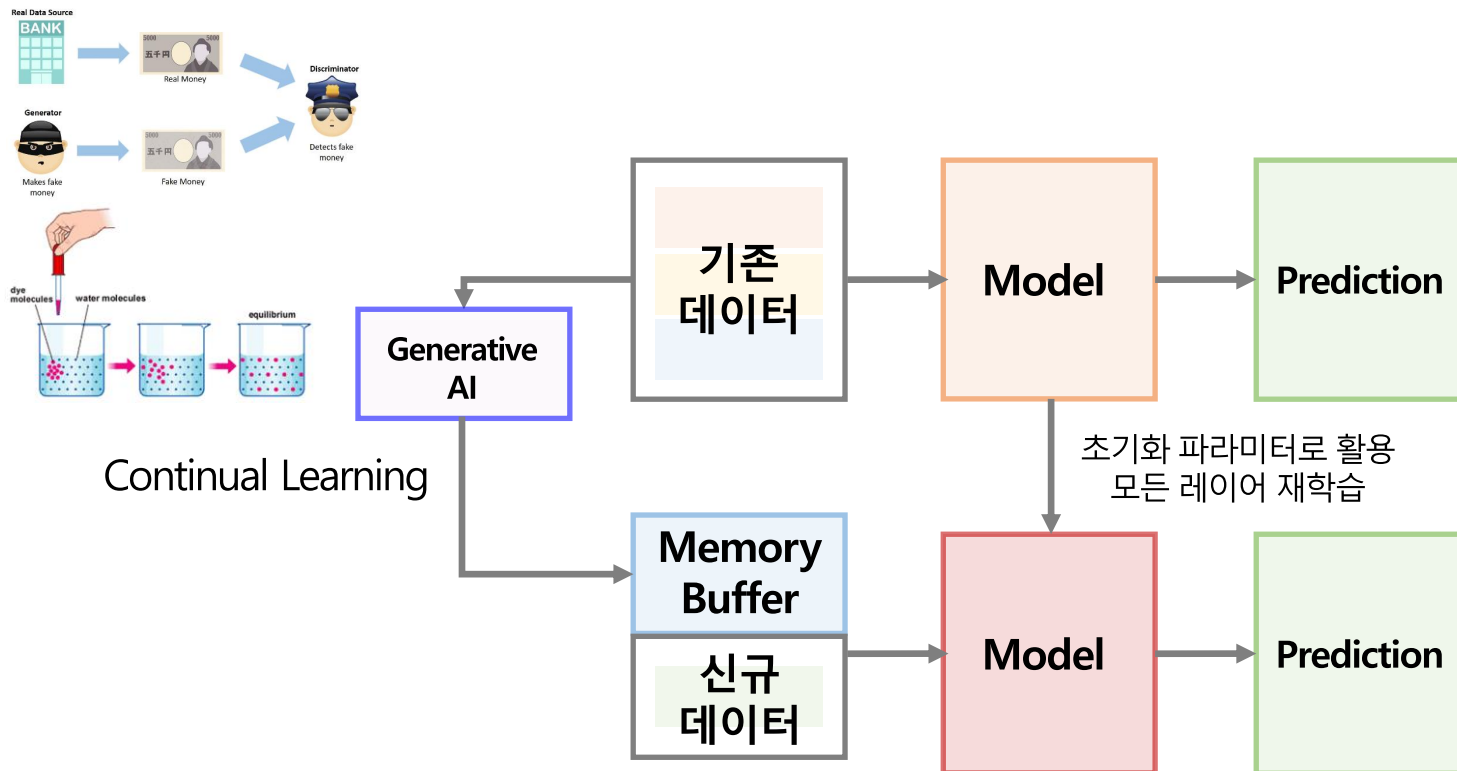


기존 데이터 정보를 잊는  
**Catastrophic Forgetting 방지!**  
기존 데이터와 신규 데이터  
모두 잘 맞추는 모델

## 2. Applications of Generative Replay in Image Data

### ❖ Generative Replay Methods

- 데이터 일반화, 보안, 메모리 측면에서 장점 존재하는 생성형 모델 연구가 활발히 이루어지는 중
- 이미지 분야에 적용된 GAN 기반 방법론과 Diffusion 기반 방법론 소개



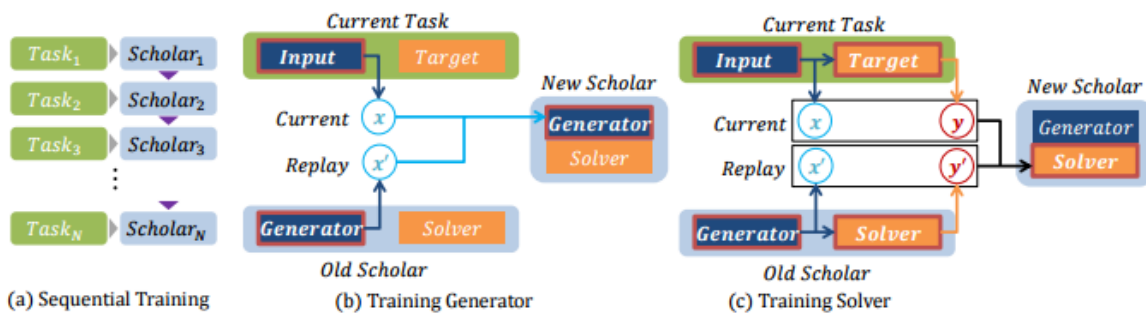
기존 데이터 정보를 잊는  
**Catastrophic Forgetting 방지!**  
기존 데이터와 신규 데이터  
모두 잘 맞추는 모델

# 2. Applications of Generative Replay in Image Data

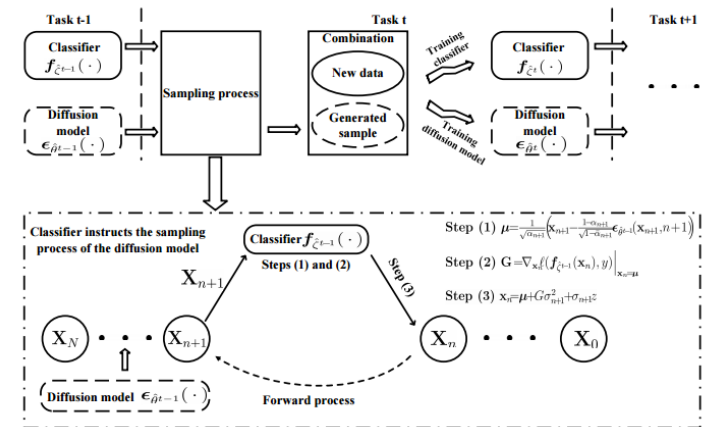
## ❖ Generative Replay Methods

- 데이터 일반화, 보안, 메모리 측면에서 장점 존재하는 생성형 모델 연구가 활발히 이루어지는 중
- 이미지 분야에 적용된 GAN 기반 방법론과 Diffusion 기반 방법론 소개

DGR, GAN을 이용한 Generative Replay



DDGR, Diffusion 모델을 활용한 Generative Replay



## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

- Generator와 Solver 두가지 개념의 모델을 활용하여 Continual Learning을 수행
- Generator에서 GAN 기법을 통해 Replay에 이용 할 데이터를 생성
- 해마(단기기억)와 신피질(장기기억) 관점에서 인간이 그대로 기억하는 것이 아니라 자신의 지식기반으로 기억을 생성하기 때문에 generative model이 더 잘 어울린다고 주장

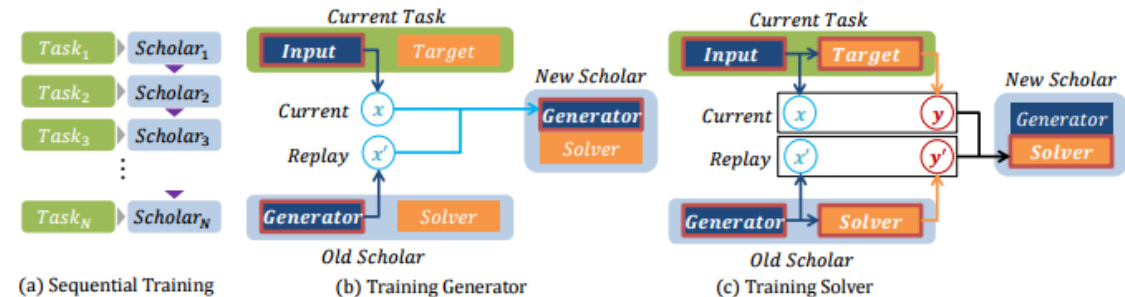
### Continual Learning with Deep Generative Replay

Hanul Shin  
Massachusetts Institute of Technology  
SK T-Brain  
skyshin@mit.edu

Jung Kwon Lee\*, Jaehong Kim\*, Jiwon Kim  
SK T-Brain  
{jklee, xhark, jk}@sktbrain.com

#### Abstract

Attempts to train a comprehensive artificial intelligence capable of solving multiple tasks have been impeded by a chronic problem called catastrophic forgetting. Although simply replaying all previous data alleviates the problem, it requires large memory and even worse, often infeasible in real world applications where the access to past data is limited. Inspired by the generative nature of the hippocampus as a short-term memory system in primate brain, we propose the Deep Generative Replay, a novel framework with a cooperative dual model architecture consisting of a deep generative model ("generator") and a task solving model ("solver"). With only these two models, training data for previous tasks can easily be sampled and interleaved with those for a new task. We test our methods in several sequential learning settings involving image classification tasks.

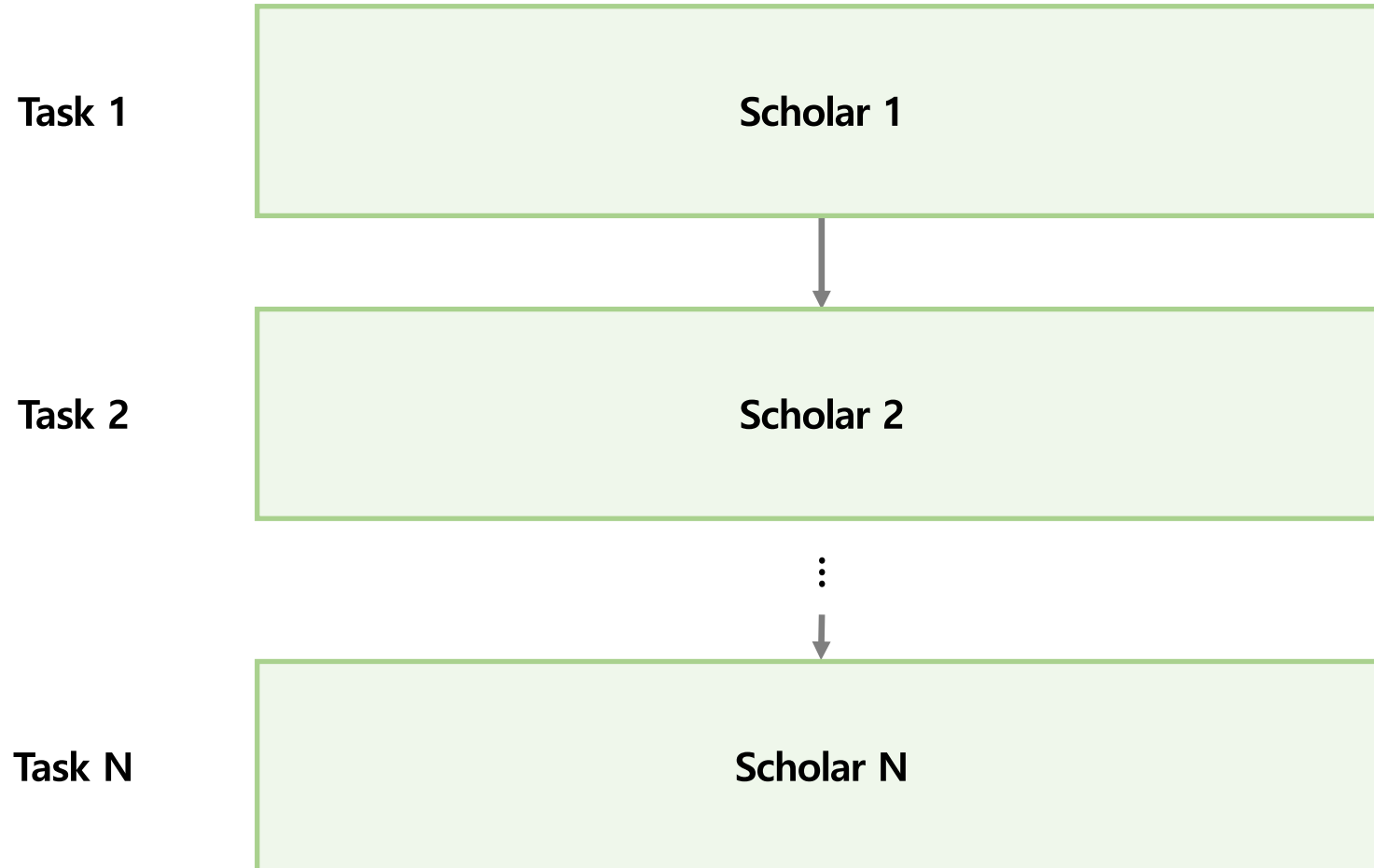


## 2. Applications of Generative Replay in Image Data

---

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

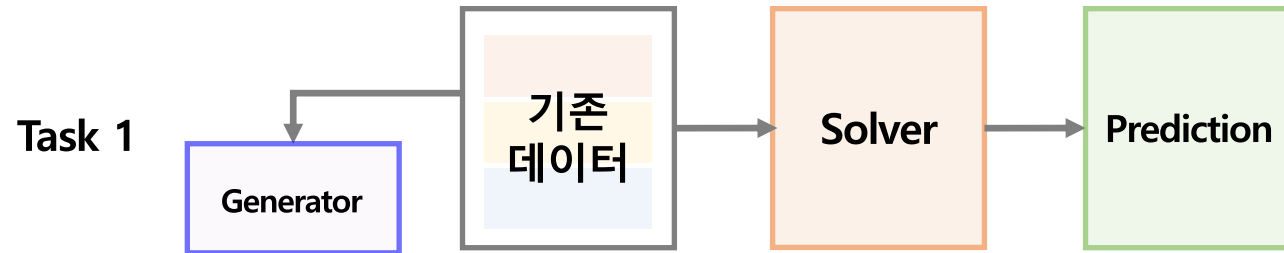
- 여러가지 Task를 Continual Learning 수행, Task 마다 Scholar 존재



## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

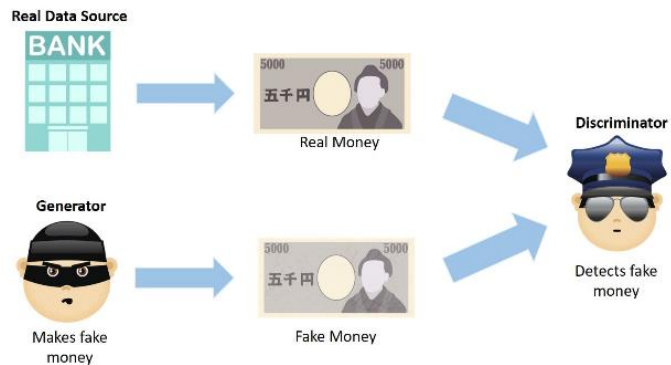
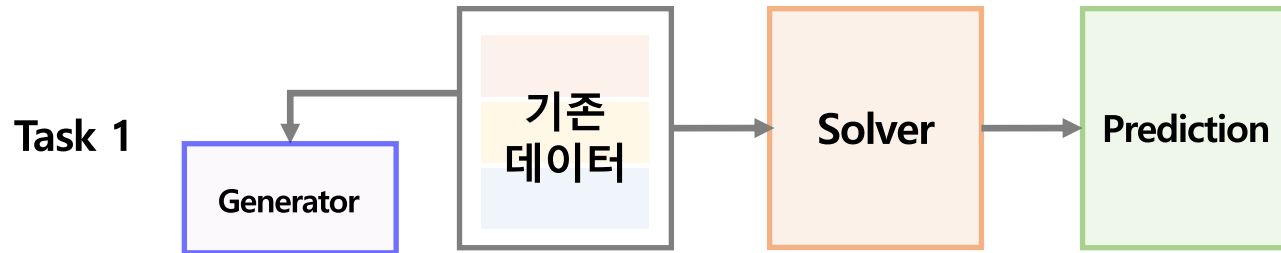
- Scholar는 Generator와 Solver로 구성



## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

- Scholar는 Generator와 Solver로 구성, Generator는 WGAN-GP 모델 활용

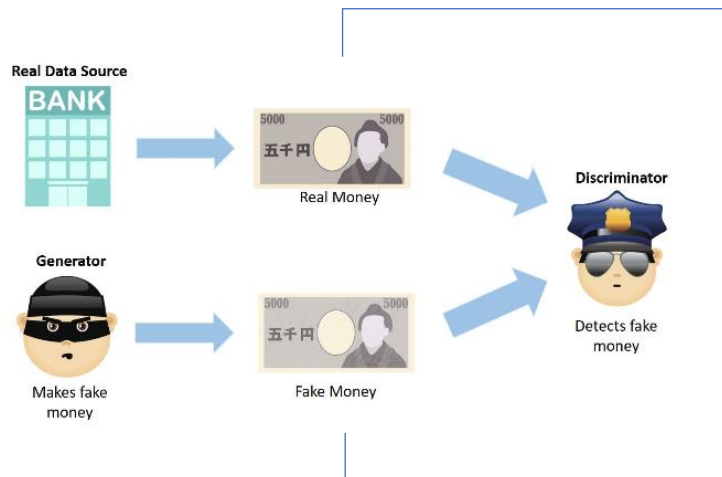
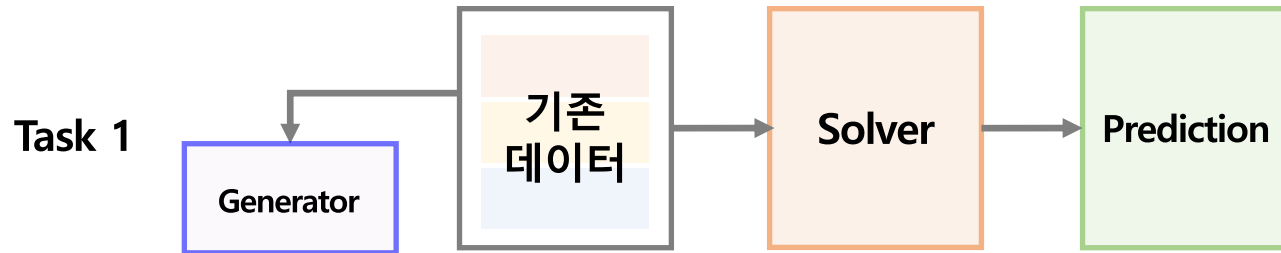




## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

- Scholar는 Generator와 Solver로 구성, Generator는 WGAN-GP 모델 활용



#### Wasserstein Distance

실제 데이터와 생성된 데이터 간의 분포 차이를 줄이는 방식으로 학습

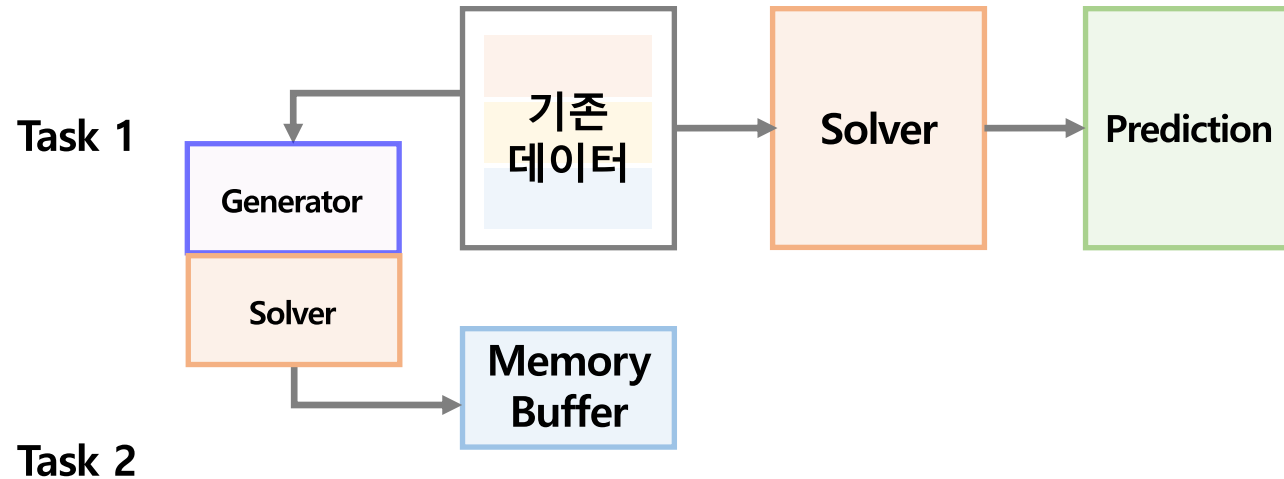
#### Gradient Penalty

기존 WGAN은 연속성 보장을 위해 가중치의 범위를 특정하여 제대로 생성이 안되는 경우 발생  
→ 범위를 특정하는 가중치 클리핑이 아닌 가중치 패널티 활용

## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

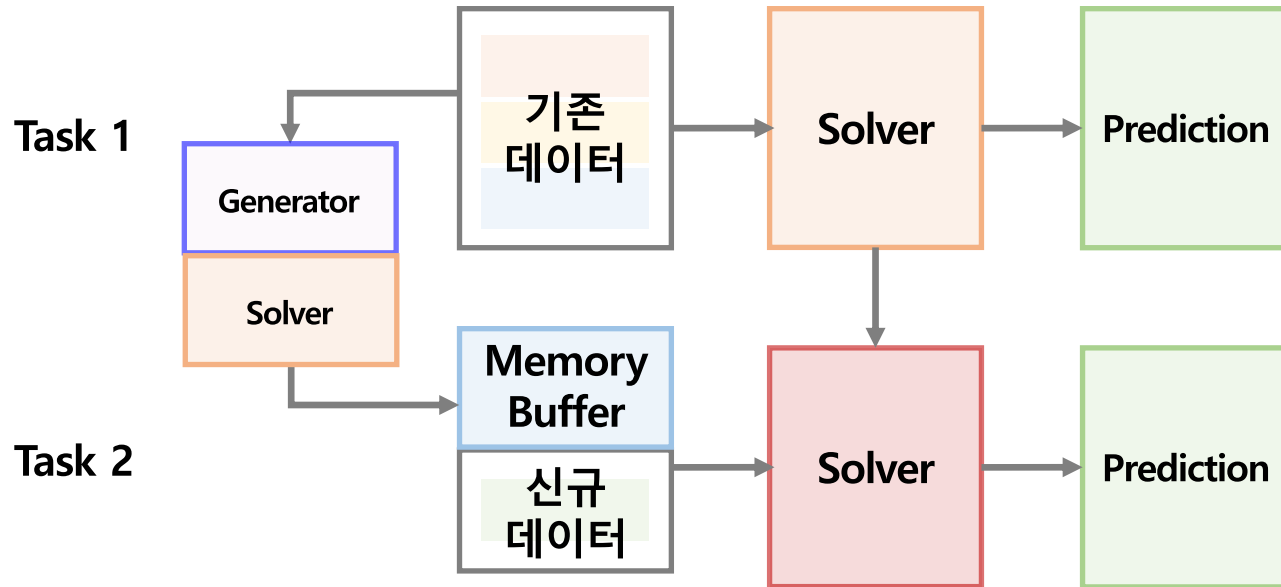
- Generator로 부터 생성된 데이터를 학습된 Solver를 이용해 label 부여



## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

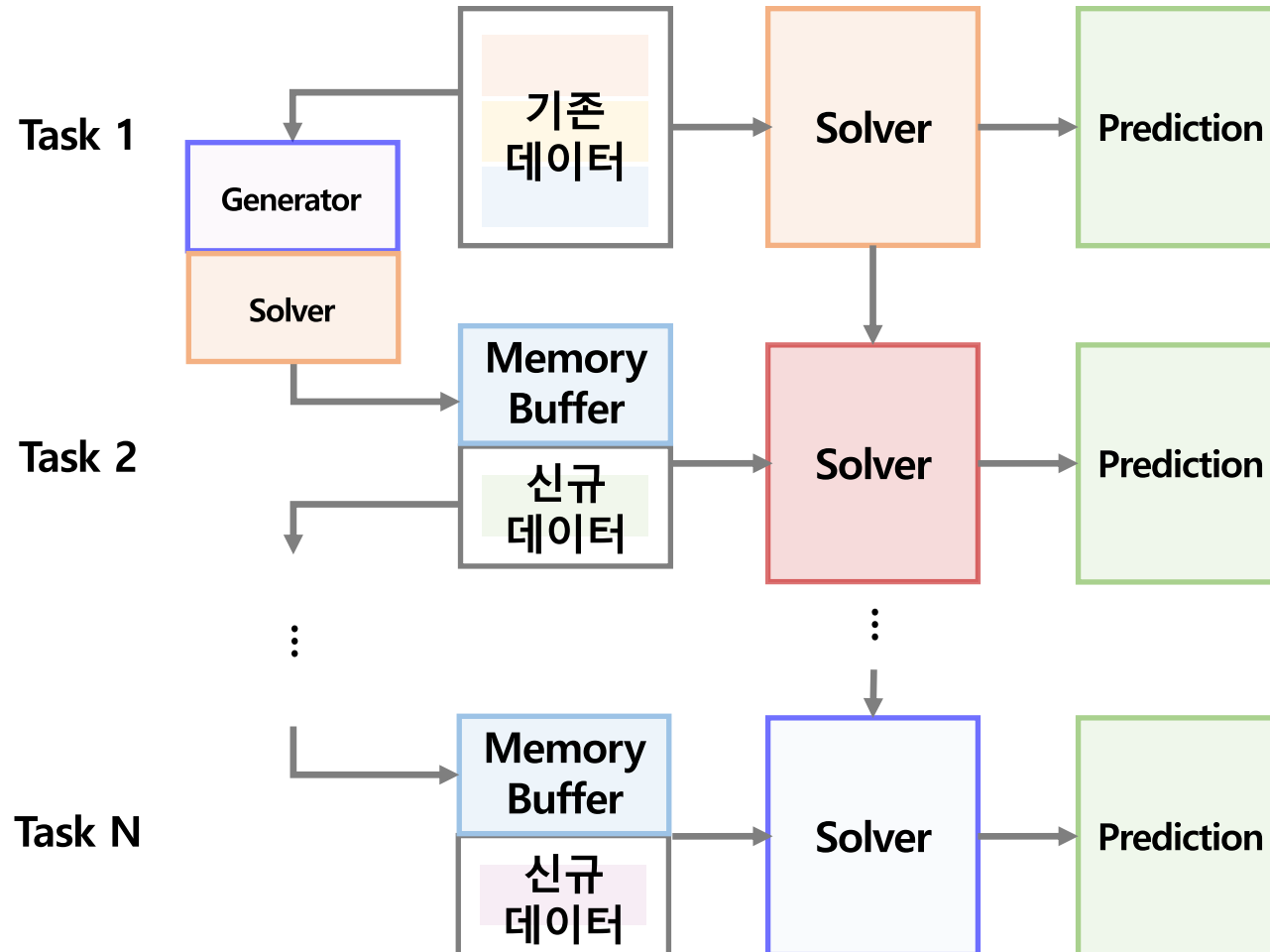
- Scholar는 Generator와 Solver로 구성, Generator는 WGAN-GP 모델 활용



## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

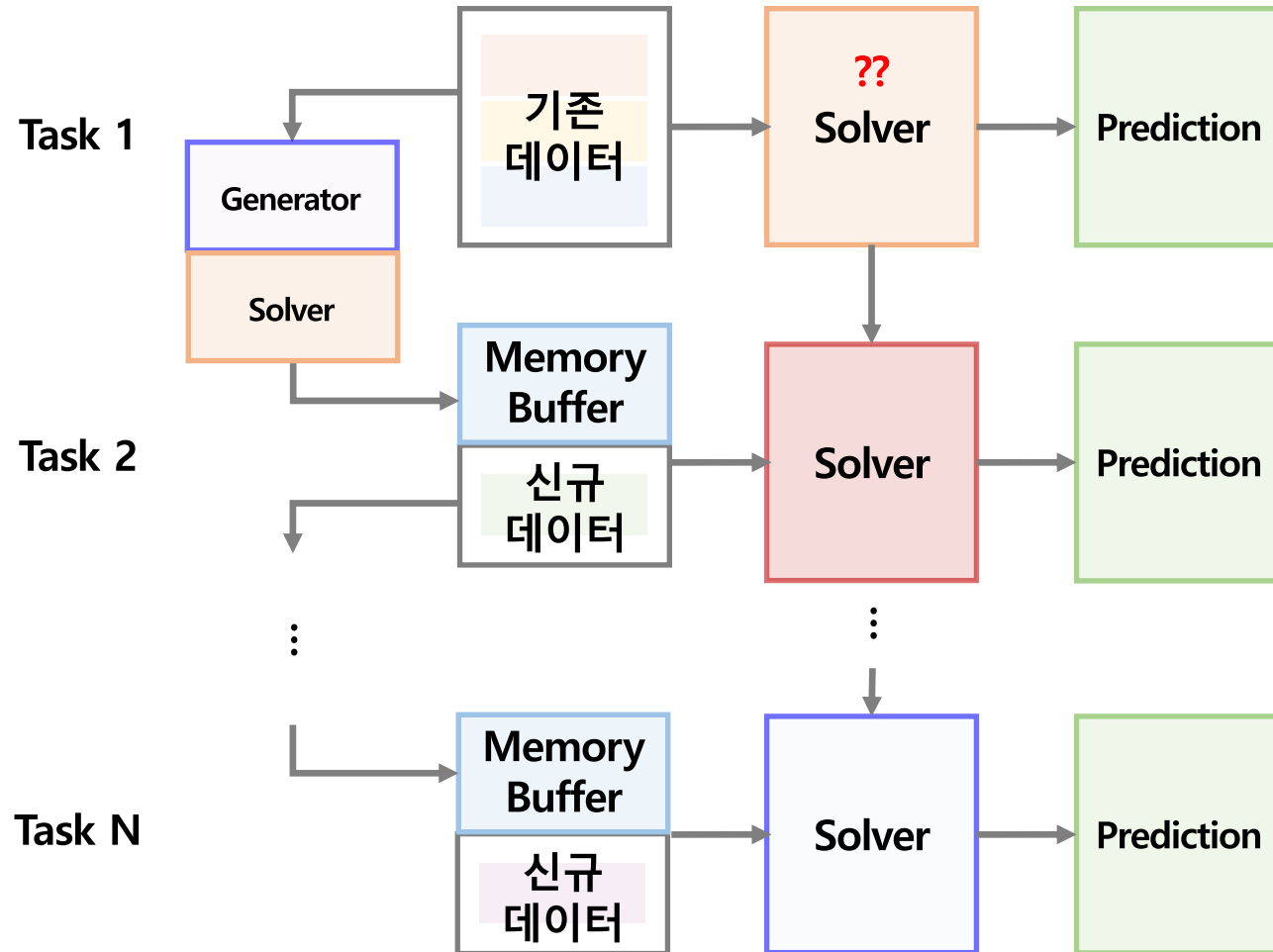
- Scholar는 Generator와 Solver로 구성, Generator는 WGAN-GP 모델 활용



## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

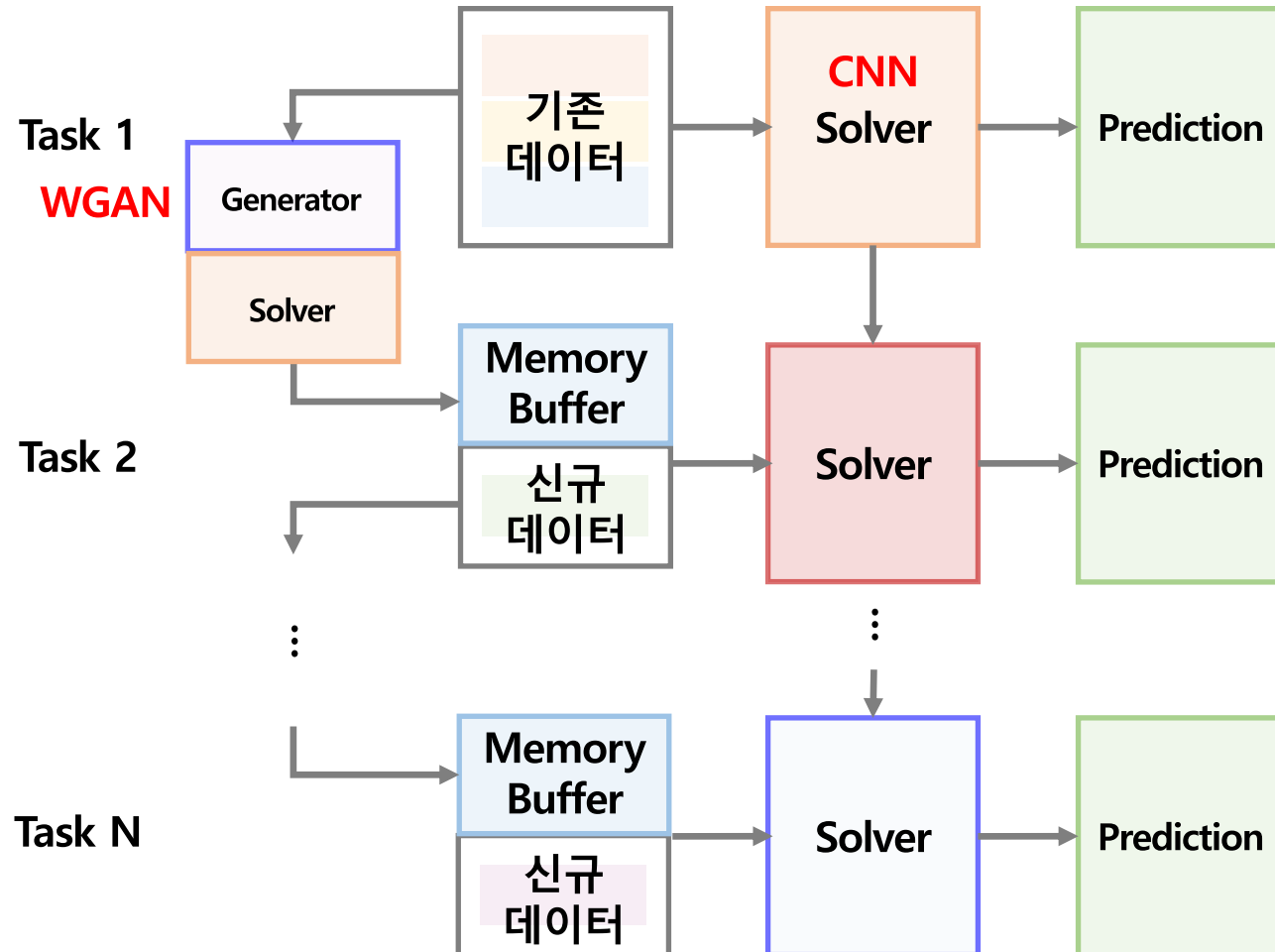
- 논문에는 구체적으로 어떤 모델을 Solver로 사용했는지 언급하지 않음



## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

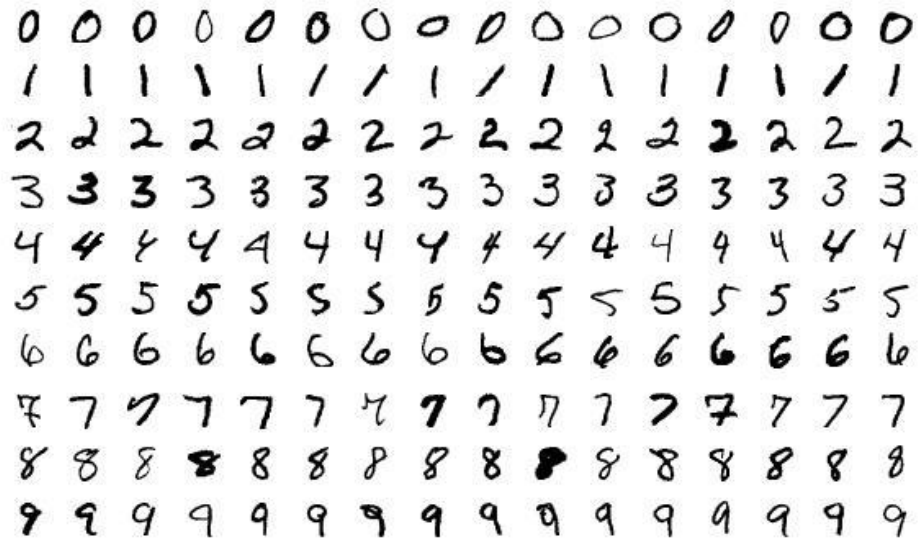
- 해당 논문을 구현한 Github에서 Generator는 WGAN, Solver는 CNN 모델 활용 <https://github.com/kuc2477/pytorch-deep-generative-replay>



## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

- 실험 데이터 소개
  - ✓ MNIST 손글씨 데이터
  - ✓ SVHN(street view house number) 건물에 써있는 글씨 데이터



## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

- MNIST 데이터 실험 결과
  - ✓ Solver 1은 실제 데이터로만 학습 수행, 이후 생성된 데이터 함께 활용
  - ✓ 지속적으로 성능이 잘 나오는 것을 확인

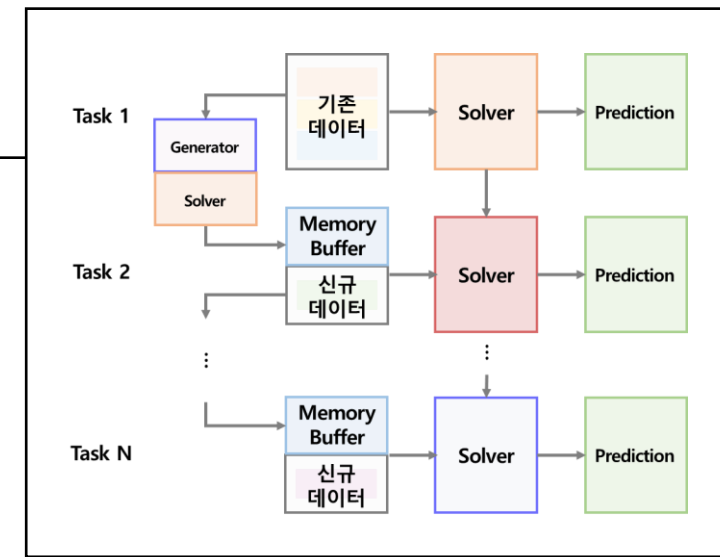
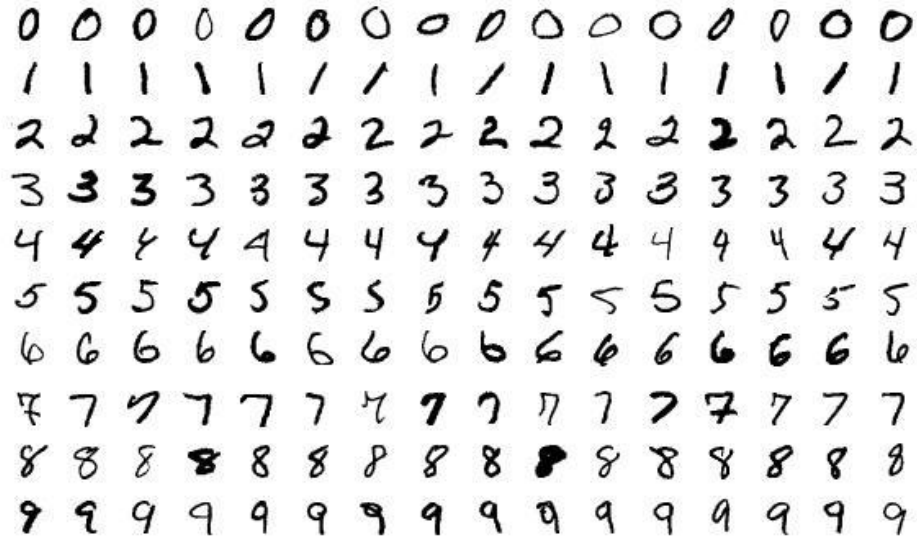


Table 1: Test accuracy of sequentially learned solver measured on full test data from MNIST database. The first solver learned from real data, and subsequent solvers learned from previous scholar networks.

	$Solver_1$	$\rightarrow$	$Solver_2$	$\rightarrow$	$Solver_3$	$\rightarrow$	$Solver_4$	$\rightarrow$	$Solver_5$
Accuracy(%)	98.81%		98.64%		98.58%		98.53%		98.56%



## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

- MNIST와 SVHN를 순차적으로 Continual Learning 한 결과
  - ✓ Experience Replay (ER) : 이전 Task의 실제 데이터를 지속적으로 추가해서 사용
  - ✓ Generative Replay (GR) : 이전 Task에서 생성된 데이터를 지속적으로 추가해서 사용
  - ✓ None : 이전 Task의 정보 없이 새로운 데이터로만 Fine-tuning
- 생성된 데이터를 이용한 GRO이 실제 데이터를 이용한 ER만큼 우수한 성능을 보임

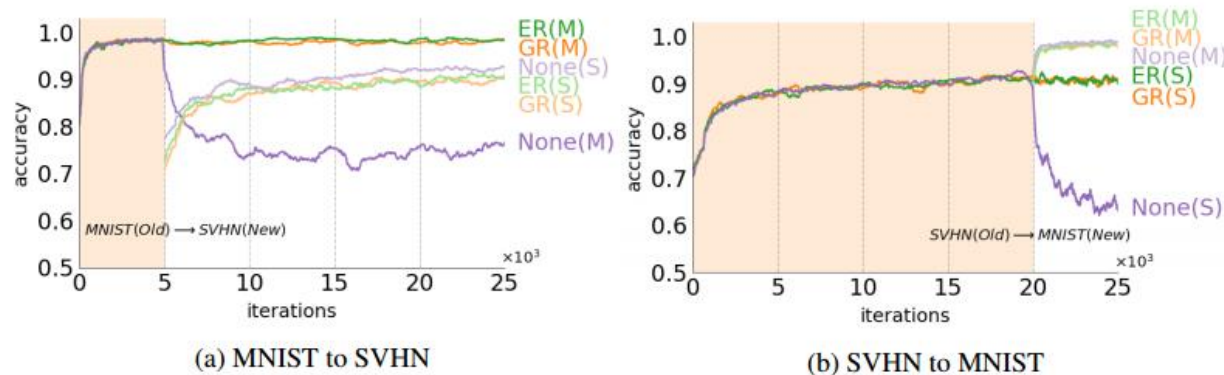


Figure 3: Accuracy on classifying samples from two different domains. (a) The models are trained on MNIST then on SVHN dataset or (b) vice versa. When the previous data are recalled by generative replay (orange), knowledge of the first domain is retained as if the real inputs with predicted responses are replayed (green). Sequential training on the solver alone incurs forgetting on the former domain, thereby resulting in low average performance (violet).

## 2. Applications of Generative Replay in Image Data

### ❖ DGR : Continual learning with deep generative replay (NIPS, 2017)

- 실제 생성된 데이터 샘플
  - ✓ MNIST 데이터는 대부분 숫자의 형태를 띄고 있지만 SVHN 데이터는 뭉개진 경우도 다수 발견
  - ✓ 데이터의 난이도 차이로 인한 결과



Figure 4: Samples from trained generator in MNIST to SVHN experiment after training on SVHN dataset for 1000, 2000, 5000, 10000, and 20000 iterations. The samples are diverted into ones that mimic either SVHN or MNIST input images.

## 2. Applications of Generative Replay in Image Data

### ❖ DDGR : Continual learning with deep diffusion-based generative replay (ICML, 2023)

- 앞서 설명했던 DGR이랑 동일한 Generative Replay Continual Learning 구조에 Diffusion 적용한 논문
- DGR의 경우 데이터가 복잡해질수록 제대로 생성 못하는 문제 발생
- 구체적인 실험 구성을 통한 다양한 알고리즘 비교

#### DDGR: Continual Learning with Deep Diffusion-based Generative Replay

Rui Gao<sup>1</sup> Weiwei Liu<sup>1</sup>

##### Abstract

Popular deep-learning models in the field of image classification suffer from catastrophic forgetting—models will forget previously acquired skills when learning new ones. Generative replay (GR), which typically consists of a generator and a classifier, is an efficient way to mitigate catastrophic forgetting. However, conventional GR methods only focus on a single instruction relationship (generator-to-classifier), where the generator synthesizes samples for previous tasks to instruct the training of the classifier, while ignoring the ways in which the classifier can benefit the generator. In addition, most generative replay methods typically reuse the generated samples to

that existing standard deep learning methods quickly forget their previously acquired experiences when learning new tasks (Kirkpatrick et al., 2017). This phenomenon, referred to as catastrophic forgetting (Kumaran et al., 2016), represents a significant challenge for scenarios in which tasks are learned in sequence and previous training data cannot be obtained.

Continual learning (CL)—also referred to as lifelong learning (Chen & Liu, 2018), sequential learning (Aljundi et al., 2019), and incremental learning (Aljundi et al., 2018)—has emerged as an efficient way to mitigate catastrophic forgetting in deep learning models. Classical CL works have explored a range of approaches to help the model remember previous knowledge. Among them, the generative replay (GR) model is one of the most popular methods due to its

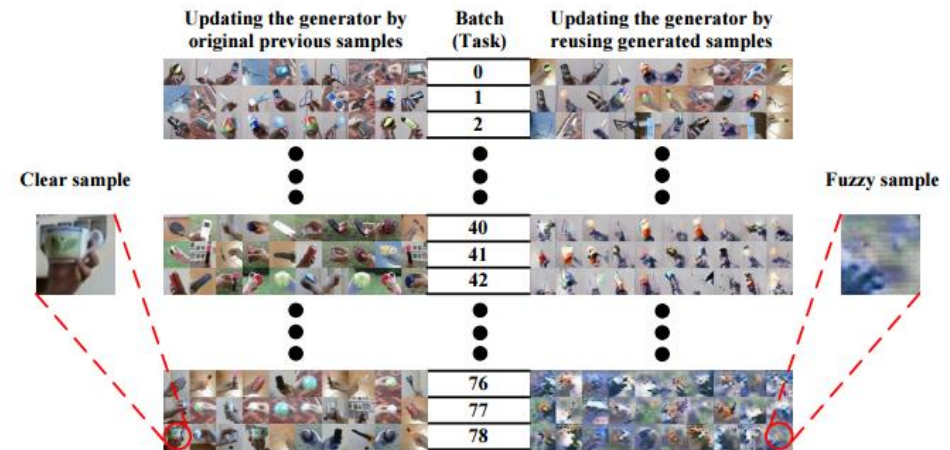
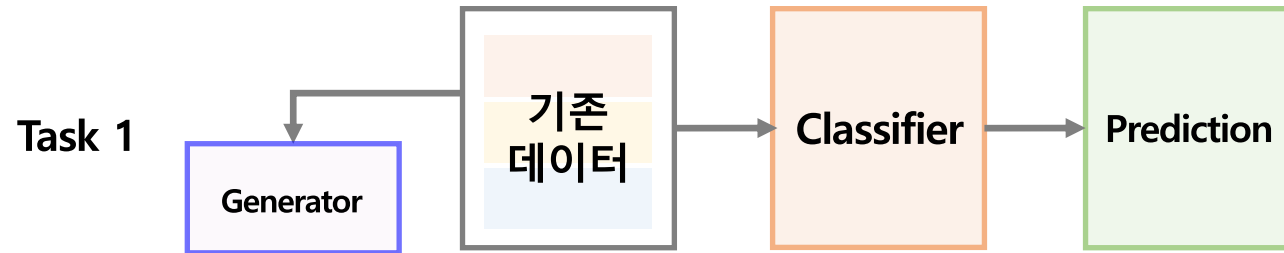


Figure 1. We use DGR as an example to demonstrate that reusing generated samples will lead to the generation of low-quality samples for previous tasks. We conduct experiments on CORE50. Further details of these experiments are provided in Section 3.2.

## 2. Applications of Generative Replay in Image Data

### ❖ Continual learning with deep diffusion-based generative replay

- Task 마다 Diffusion model과 Classifier로 구성



# 2. Applications of Generative Replay in Image Data

## ❖ Continual learning with deep diffusion-based generative replay

- Classifier는 CNN 계열 모델인 AlexNet과 ResNet 활용

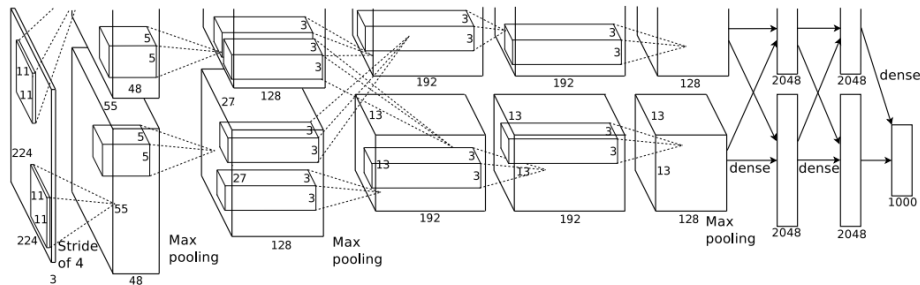
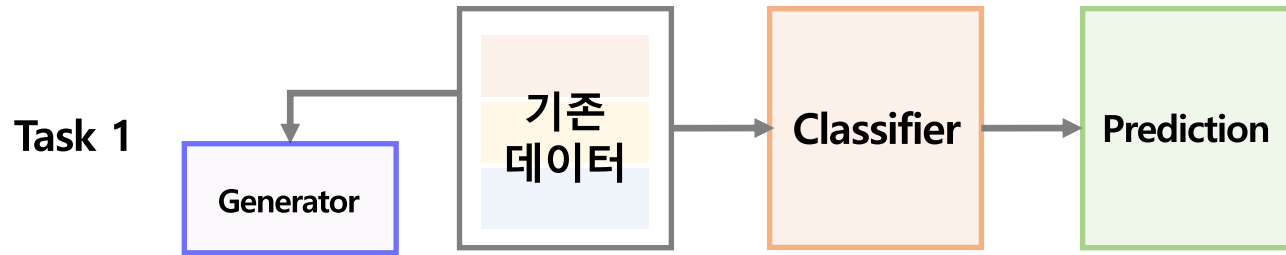
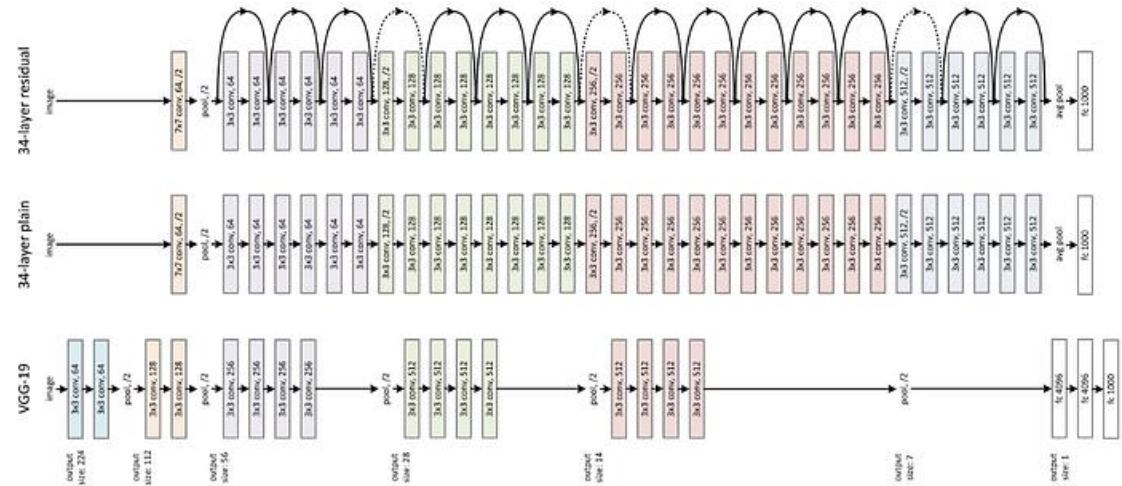


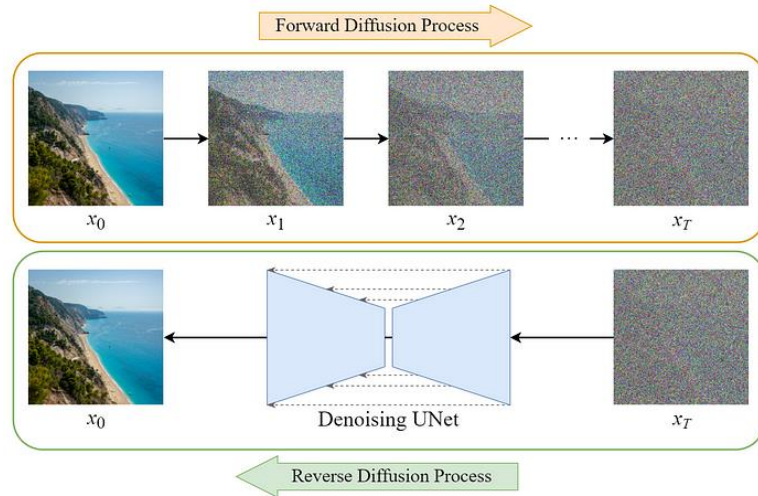
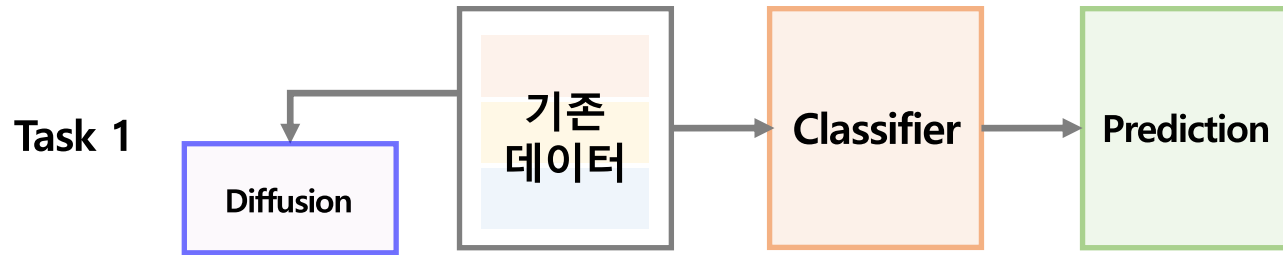
Figure 2: An illustration of the architecture of our CNN, explicitly showing the delineation of responsibilities between the two GPUs. One GPU runs the layer-parts at the top of the figure while the other runs the layer-parts at the bottom. The GPUs communicate only at certain layers. The network's input is 150,528-dimensional, and the number of neurons in the network's remaining layers is given by 253,440–186,624–64,896–64,896–43,264–4096–4096–1000.



## 2. Applications of Generative Replay in Image Data

### ❖ Continual learning with deep diffusion-based generative replay

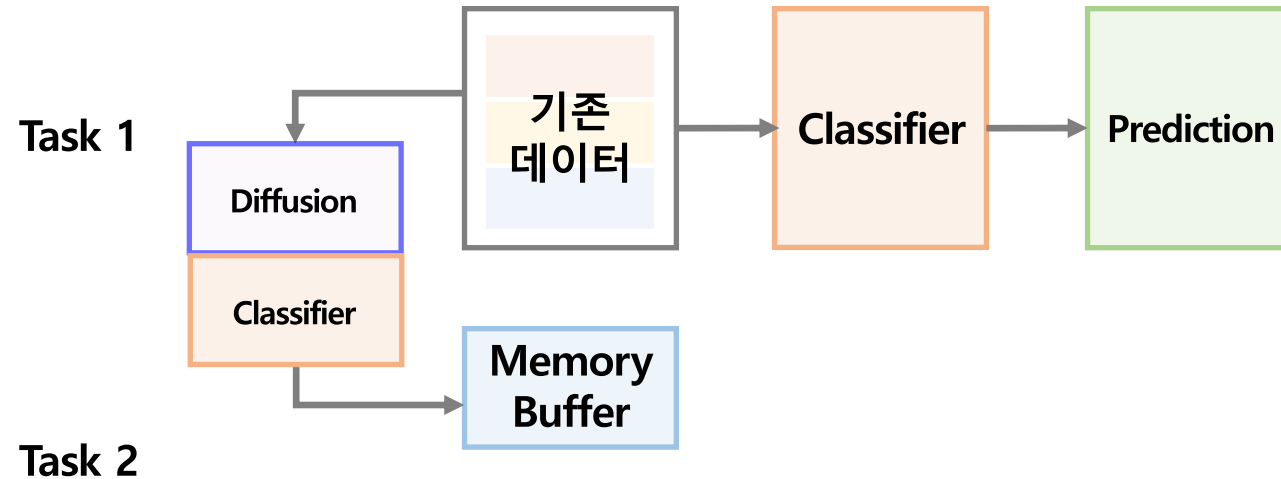
- Diffusion model은 DDPM 활용



## 2. Applications of Generative Replay in Image Data

### ❖ Continual learning with deep diffusion-based generative replay

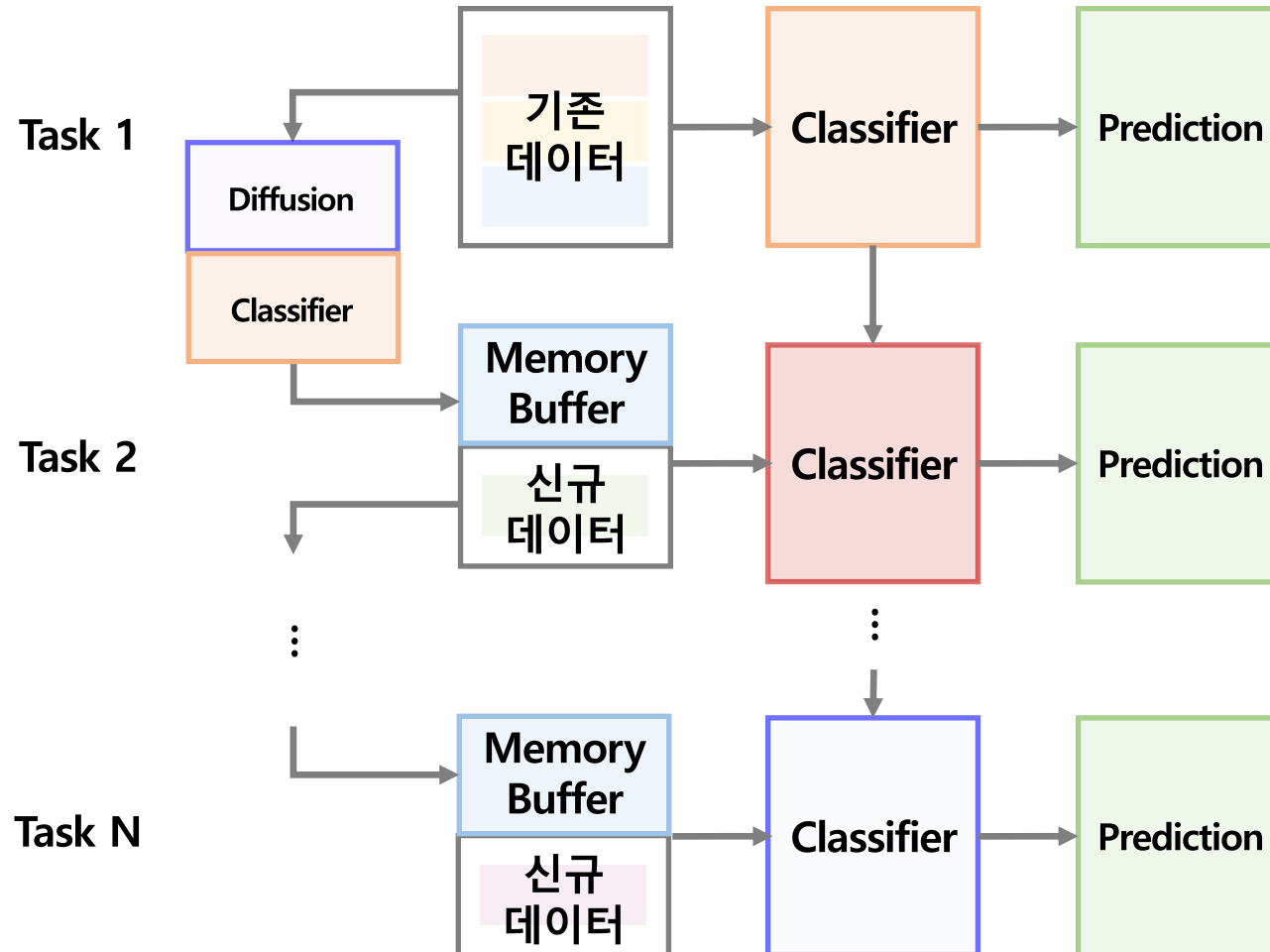
- Diffusion model로 부터 생성된 데이터를 학습된 Classifier로 평가하여 label 부여



## 2. Applications of Generative Replay in Image Data

### ❖ Continual learning with deep diffusion-based generative replay

- Task 마다 Diffusion model과 Classifier로 구성

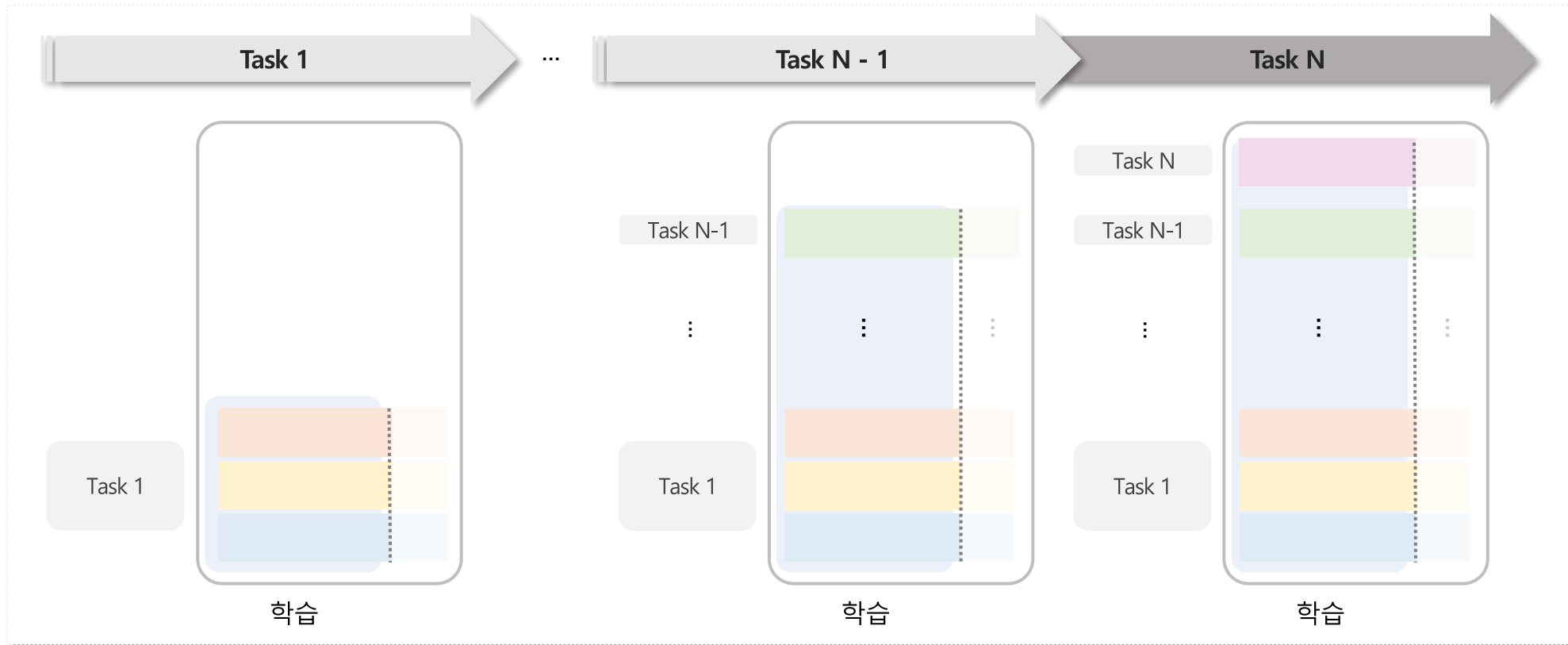




## 2. Applications of Generative Replay in Image Data

### ❖ Continual learning with deep diffusion-based generative replay

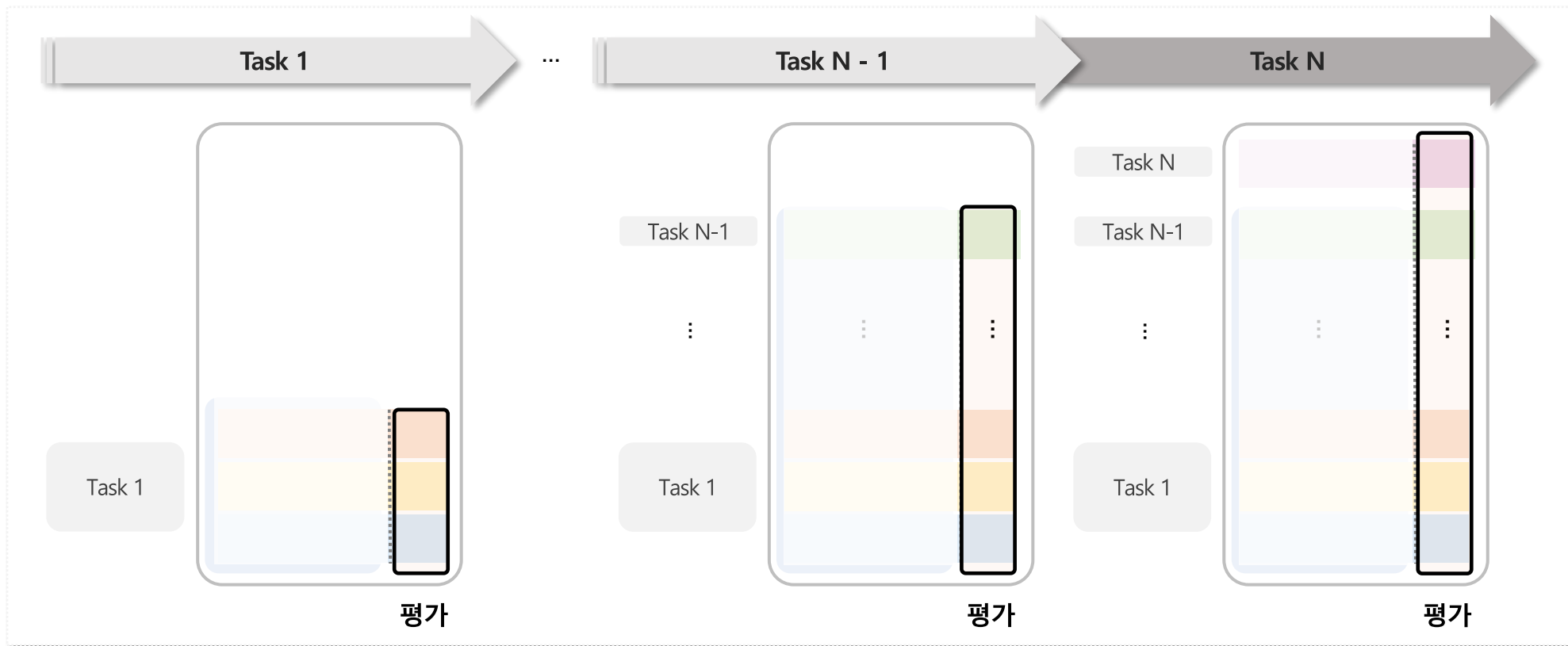
- 각 모델에 대하여 N개의 Task로 구성
- (1) Phase 2에 대한 평가 성능과 (2) Task 간 사전 학습 데이터셋에 대한 평가 성능 비교 (Forgetting 측정)



## 2. Applications of Generative Replay in Image Data

### ❖ Continual learning with deep diffusion-based generative replay

- 각 모델에 대하여 N개의 Task로 구성
- (1) Phase 2에 대한 평가 성과와 (2) Task 간 사전 학습 데이터셋에 대한 평가 성능 비교 (Forgetting 측정)



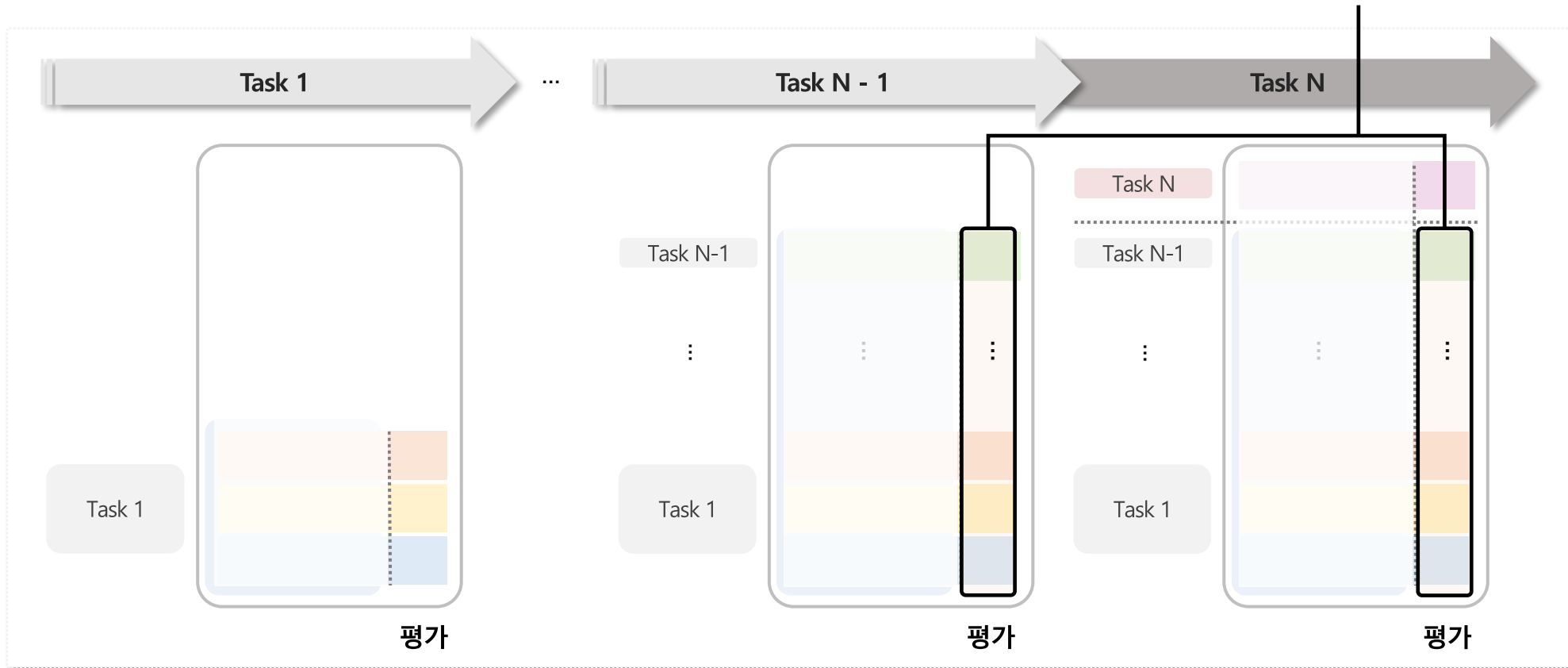
$$A_T = acc_{T,0:T}$$

각 Task 정확도 평균

## 2. Applications of Generative Replay in Image Data

### ❖ Continual learning with deep diffusion-based generative replay

- 각 모델에 대하여 N개의 Task로 구성
- (1) Phase 2에 대한 평가 성과와 (2) Task 간 사전 학습 데이터셋에 대한 평가 성능 비교 (Forgetting 측정)



$$F_{avg}^T = \frac{\sum_{c \in C_T} F_c^T}{|C_T|}$$

직전 Task와의  
Forgetting 차이 평균

## 2. Applications of Generative Replay in Image Data

### ❖ Continual learning with deep diffusion-based generative replay

- 실험 데이터 소개
  - ✓ CIFAR100
  - ✓ ImageNET

airplane



automobile



bird



cat



deer



dog



frog



horse



ship



truck



## 2. Applications of Generative Replay in Image Data

### ❖ Continual learning with deep diffusion-based generative replay

- 단순 Fine-tuning, 모델 관점 CL, 메모리 관점 CL 모델 실험 수행
- 각각 5번의 Task 수행, NC는 Class의 수를 의미
  - ✓ CIFAR 100의 절반 Class 데이터를 Task 1에서 활용하고 이후에 5개 또는 10개 Class를 추가 Task마다 활용
  - ✓ ImageNet 중 500개 Class 데이터를 Task 1에서 활용하고 이후에 50개 또는 100개 Class를 추가 Task마다 활용

Table A.1. Details of datasets

	Tasks	Classes/task	Train data/task	Task selection
CIFAR-100	5	{50, 5} or {50, 10}	{25000, 2500} or {25000, 5000}	random class
ImageNet	5	{500, 50} or {500, 100}	{650000, 65000} or {650000, 130000}	random class

Table 1. All results on two datasets in the CI scenario. We present the final average accuracy  $A_T$  and forgetting rate  $F_T^{avg}$ .

Method	$A_T = acc_{T,0:T}$								$F_T^{avg} = \sum_{c \in C_T} F_T^c /  C_T $							
	CIFAR-100				ImageNet				CIFAR-100				ImageNet			
	AlexNet		ResNet		AlexNet		ResNet		AlexNet		ResNet		AlexNet		ResNet	
	NC=5	10	5	10	50	100	50	100	5	10	5	10	50	100	50	100
Finetuning	6.11	5.12	18.08	17.50	5.33	3.24	12.95	10.28	60.45	59.87	61.65	62.79	56.55	57.83	58.58	59.71
SI	16.96	13.57	26.45	23.15	19.38	14.38	28.88	24.38	48.58	50.18	52.27	56.65	41.18	45.94	41.93	44.56
EWC	15.29	9.71	25.49	18.82	15.22	13.03	23.51	22.03	50.38	54.27	52.83	60.47	45.65	47.01	46.98	46.96
MAS	20.13	18.94	29.94	28.28	16.35	14.51	31.25	25.51	45.75	45.37	49.38	51.31	44.85	45.70	39.55	44.34
IMM	11.26	9.87	21.02	19.79	13.68	11.13	23.19	19.73	54.60	54.57	58.12	59.89	46.65	49.31	47.70	49.62
DGR	42.49	38.16	52.96	48.94	43.94	38.81	53.32	47.56	24.08	26.52	26.36	31.14	17.31	22.52	17.96	21.84
MeRGAN	46.03	43.23	57.19	55.69	—	—	—	—	36.95	26.49	20.12	22.55	—	—	—	—
PASS	53.21	48.65	62.30	<b>60.63</b>	—	—	—	—	27.35	19.43	16.97	21.21	—	—	—	—
DDGR	<b>59.20</b>	<b>52.22</b>	<b>63.40</b>	60.04	<b>53.86</b>	<b>52.21</b>	<b>64.83</b>	<b>61.26</b>	<b>23.00</b>	<b>16.86</b>	<b>15.34</b>	<b>19.25</b>	<b>6.98</b>	<b>7.82</b>	<b>5.65</b>	<b>7.73</b>

모델 관점 CL

데이터 관점 CL

## 2. Applications of Generative Replay in Image Data

### ❖ Continual learning with deep diffusion-based generative replay

- 실제 생성된 데이터 샘플
  - ✓ 기존 GAN을 이용한 DGR의 경우 이미지가 어렵고 Task가 길어질수록 정상적인 생성이 어려움
  - ✓ DDGR의 경우 이미지 생성도 잘 해내는 모습

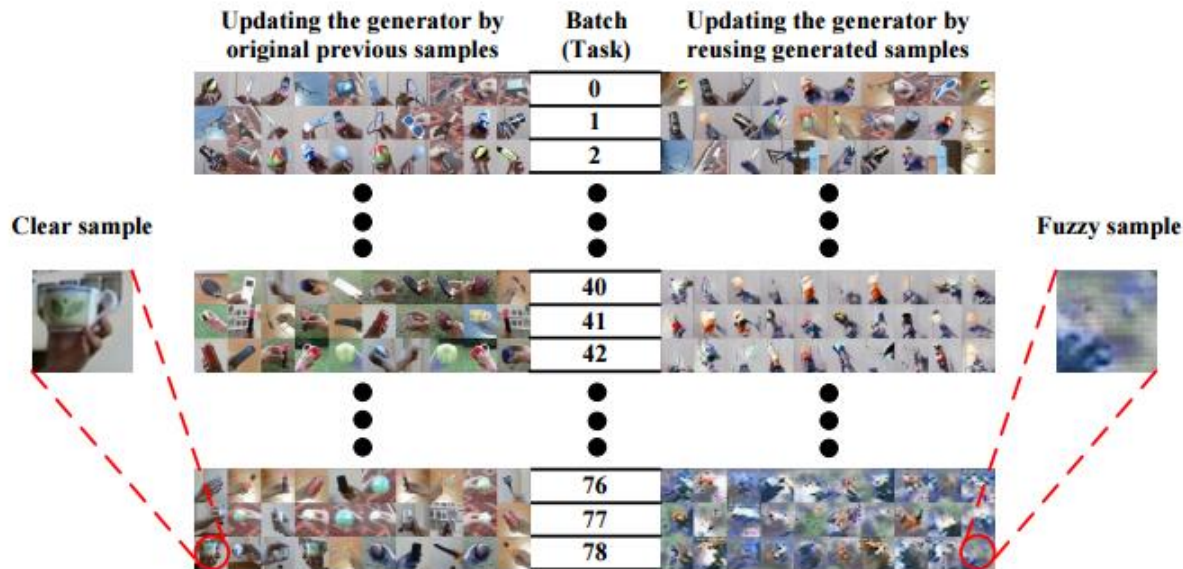
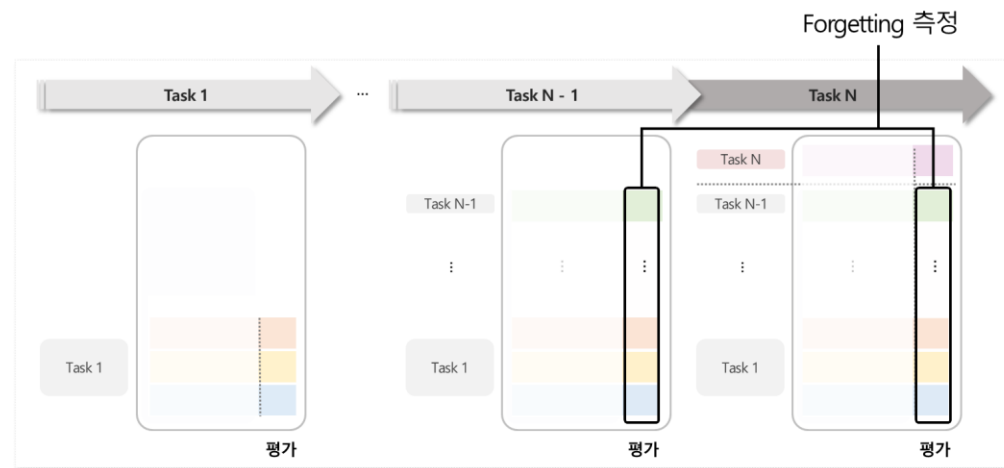
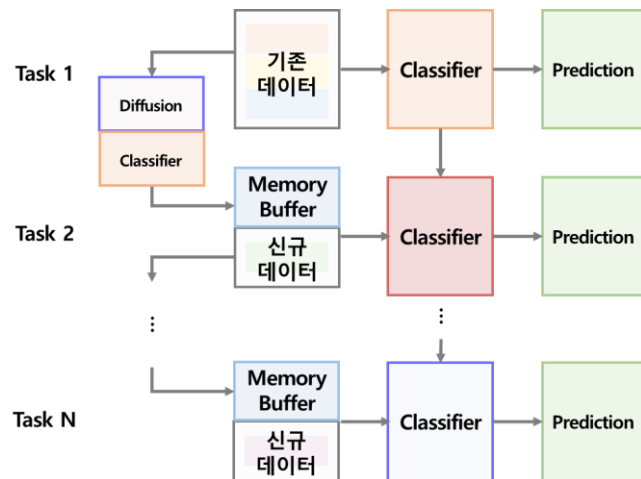


Figure E.1. The samples generated by DDGR at the last three batches.

# 3. Conclusion

## ❖ Summary

- Continual Learning은 이전에 학습한 내용을 잃지 않고 유지하는 학습 방법
  - ✓ 모델이 변화하는 데이터 환경에서 지속적으로 업데이트하여 데이터가 점진적으로 추가 되거나 변경 될 때 유용 / Class, Domain incremental
  - ✓ Fine-tuning을 기준으로 **Catastrophic Forgetting**을 해결하는 것이 목표 (Transfer learning은 유사한 데이터 특성 등 특정 조건 필요)
  - ✓ 기존 데이터를 보관 할 수 없는 경우를 생각하여 기존 데이터의 일부만 활용하거나 활용하지 않음
- DGR, DDGR 등 GAN과 Diffusion을 이용한 생성 모델 기반 Continual Learning 리뷰
  - ✓ DGR : WGAN을 이용하여 이전 Task의 실제 데이터로부터 학습한 분포를 이용해 다음 Task 학습에 사용 할 데이터 생성
  - ✓ DDGR : DDPM을 이용하여 이전 Task의 실제 데이터로부터 학습한 분포를 이용해 다음 Task 학습에 사용 할 데이터 생성



고맙습니다